2004-09

Information Security and Wireless alternate approaches for controlling access to critical information

Nandram, Winsome

Monterey, California. Naval Postgraduate School

NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

INFORMATION SECURITY AND WIRELESS: ALTERNATE APPROACHES FOR CONTROLLING ACCESS TO CRITICAL INFORMATION

by

Winsome Nandram

September 2004

Thesis Advisor:	Gurminder Singh
Second Reader:	Arijit Das

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	September 2004	Master's Thesis

4. TITLE AND SUBTITLE: Information Security and Wireless: Alternate Approaches for Controlling Access to Critical Information	5. FUNDING NUMBERS
6. AUTHOR(S) Winsome Nandram	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A	10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT	12b. DISTRIBUTION CODE
Approved for public release; distribution is unlimited	

13. ABSTRACT (maximum 200 words)

The advent of Wireless Local Area Networking (WLAN) has seen a widespread adoption of its technology and functionality in many different areas. Many studies show more and more organizations are extending their networks to incorporate wireless devices and their applications. Permitting wireless devices to access private networks however, further complicates the tasks of protecting the network and its resources from unauthorized access. Now that they have become a significant element in today's networks, selecting and deploying adequate security measures have become the focus of many research efforts. Typically, network managers implement countermeasures to augment security.

The goal of this thesis is to research approaches that compliment existing security measures with fine grain access control measures. The Extensible Markup Language (XML) is adopted to accommodate such granular access control as it provides the mechanisms for scaling security down to the document content level.

14. SUBJECT TERMS Wireless Local Area Networking, WLAN, Networks, Wireless Devices, Extensible Markup Language, XML	15. NUMBER OF PAGES 91
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

NSN 7540-01-280-5500 Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

INFORMATION SECURITY AND WIRELESS: ALTERNATE APPROACHES FOR CONTROLLING ACCESS TO CRITICAL INFORMATION

Winsome A. Nandram
Captain, United States Marine Corps
B.A., Campbell University, 1999

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE

from the

NAVAL POSTGRADUATE SCHOOL
September 2004

Author: Winsome Nandram

Approved by: Gurminder Singh
Thesis Advisor

Arijit Das
Co-Advisor

Peter Denning
Chairman, Department of Computer Science

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The advent of Wireless Local Area Networking (WLAN) has seen a widespread adoption of its technology and functionality in many different areas. Many studies show more and more organizations are extending their networks to incorporate wireless devices and their applications. Permitting wireless devices to access private networks however, further complicates the tasks of protecting the network and its resources from unauthorized access. Now that they have become a significant element in today's networks, selecting and deploying adequate security measures have become the focus of many research efforts. Typically, network managers implement countermeasures to augment security.

The goal of this thesis is to research approaches that compliment existing security measures with fine grain access control measures. The Extensible Markup Language (XML) is adopted to accommodate such granular access control as it provides the mechanisms for scaling security down to the document content level.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

I. INTRODUCTION 1
A. BACKGROUND 1
1 802.11 Security 1
2. Extensible Markup Language 3
B. THESIS SCOPE 4
C. EXPECTED BENEFITS OF THE RESEARCH 4
D. THESIS ORGANIZATION 5

II. XML 7
A. INTRODUCTION 7
1. What is XML 7
2. The Benefits of XML 7
B. XML BASICS 8
1. The XML Document 8
a. Attribute 8
b. Tag 9
c. Element 9
d. CDATA 9
e. Data 10
2. Schema and DTD 10
3. Parsers 11
C. SECURING XML DOCUMENT CONTENT 12
D. XML AND ACCESS CONTROL 12
1. Access Control Models 13
2. Enforcement Mechanisms 13

III. DESIGN CONSIDERATIONS FOR XML SOLUTION 15
A. XML DESIGN CONSIDERATIONS 15
1. Related Work 15
a. Subjects 15
b. Objects 16
c. Actions 16
d. Conditions 17
e. Policy Specification 17
f. Authorization Architecture 18
2. XML Policy File 19
3. Transforming the Document 21
a. XSLT 21
b. Creating the XSLT Style Sheet 22
B. WIRELESS NETWORK DESIGN CONSIDERATIONS 22
1. RADIUS Overview 23
2. Practical Approaches for Wireless Security in Depth Up to the Application Level 24

C. OVERALL DESIGN AND ARCHITECTURE 26
IV. DEVELOPMENT AND IMPLEMENTATION OF THE PROTOTYPE 29
A. DEVELOPMENT METHODOLOGY 29
B. DEVELOPMENT TOOLS 29
1. XML SPY 29
2. MSXML 31
3. Java API for XML Processing 32
4. Java Servlets 33
C. SAMPLE XML FILES 34
1. Sample XML Document 34
2. Sample XSLT Style Sheet 35
D. DEMONSTRATION JAVA PROGRAM 37
1. Option 1: Non Web-Based Application 37
a. Design and Implementation 37
2. Option 2: Web-Based Application 40
E. SUMMARY 42
1. Advantages 42
2. Limitations 43
V. CONCLUSION AND FUTURE WORK 45
A. CONCLUSION 45
B. FUTURE WORK 46
1. XML Access Control Language 46
2. Prototype 46
C. RECOMMENDATIONS 47
APPENDIX A. SAMPLE XML DOCUMENT VERSION 1 49
APPENDIX B. SAMPLE XML DOCUMENT VERSION 2 57
APPENDIX C. XSL STYLE SHEET 59
APPENDIX D. SOURCE CODE DEMONSTRATION SAX PARSER 63
LIST OF REFERENCES 73
INITIAL DISTRIBUTION LIST 75

LIST OF FIGURES

Figure 1. Subject Specification.....15
Figure 2. Object Specification.....16
Figure 3. Action Specification.....16
Figure 4. Condition Specification [From Hada 2000].....17
Figure 5. Policy Specification [From Hada 2000].....18
Figure 6. Authorization Architecture [From Hada 2000].....19
Figure 7. Sample Document.....21
Figure 8. XSLT Transformation.....22
Figure 9. RADIUS Infrastructure.....24
Figure 10. Overall Architecture Design [After Hada 2000]...27
Figure 11. Screen Capture of Default Enhanced Grid View....30
Figure 12. Screen Capture of Text View.....31
Figure 13. Outline JAXP SAX parsing APIs [From JAXP 2004]..33
Figure 14. Processing Model of a Java Servlet [From Hall 2000].....34
Figure 15. View for User with “High” Rating.....36
Figure 16. View for User with “low” Rating.....37
Figure 17. Login Screen.....38
Figure 18. Demonstration FileChooser.....40
Figure 19. Application Architecture Components.....41

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

Table 1. Modifying Symbols.....9
Table 2. Summary of Actions [From Hada 2000].....17

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

I would like to thank the many people who made this thesis possible.

My advisors, you have been very supportive and professional. Professor Singh, thank you for the opportunity and for stimulating my interest in wireless networking and wireless security. Thank you, Arijit, for your patience, guidance and remarkable technical expertise. It was a pleasure working with you both.

Special thanks to my family and friends for their encouragement and support throughout this challenging process.

I would also like to express my gratitude to my invaluable editor, Nancy Sharrock. Finally, thank God from whom all blessings flow.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. BACKGROUND

The increase in use of wireless devices to access information has generated various research efforts aimed at strengthening information security. Often there is a need to restrict access to documents due to the sensitive nature of their contents. The level of restriction would depend on the sensitivity of the material in the document. Access would only be granted to users with the appropriate access credentials. The advent of wireless networking has given rise to the need to also restrict access to certain highly sensitive document content because of the type of medium used to access the document. Unlike wired networks, wireless network traffic must be considered as being delivered to the adversary as well as the intended party. Eavesdropping is a major concern because of the ease with which transmissions can be captured. Precautions, such as denying access to certain critical information if it is being accessed over the wireless network can reduce the likelihood of compromise. The purpose of this thesis is to research approaches that consider user level access rights and access medium when granting or denying access to critical information.

1 802.11 Security

The IEEE 802.11 series of standards define wireless local area networks and describes the communication that occurs within them. The standard includes the Wireless Equivalent Privacy (WEP) protocol whose primary security goal is to protect wireless communication from eavesdropping by providing data confidentiality comparable

to that of a wired local area network. The protocol, however, does not meet its fundamental goals and possesses several serious limitations. To encrypt packets, WEP uses a single secret key shared between all mobile stations and access points of the wireless LAN. Since WEP does not employ a key management technique, key compromises are often overlooked. The most serious limitation of WEP however, is its susceptibility cryptanalysis. For encryption, WEP uses the stream cipher RC4. Stream ciphers typically expand a secret key into an arbitrarily pseudo-random "keystream" that is XORed with the plaintext for encryption and ciphertext for decryption. In the case of WEP, a secret key is concatenated with a 24-bit public Initialization Vector (IV) that is different for each packet, to generate its keystream. This design has several inherent problems. Since the WEP's IV uses only 24 bits, it is highly likely that the same IV will be reused for multiple messages. Encrypting two messages with the same IV and key can reveal information about each message. Once the duplication is discovered, various methods can be used to recover their contents. Additionally, the passive eavesdropper who can obtain several encrypted packets whose first byte of plaintext is known can deduce the secret key by exploiting the properties of the RC4 key schedule [Arbaugh 2003].

Despite their open nature and the limitations of 802.11, wireless networks have quickly become extremely popular and are continuing to grow. Until the wireless network is at least as secure as the wired, people will continue to focus on viable alternative approaches that utilize current technology.

2. Extensible Markup Language

Restricting or granting access to whole documents based on access rights and sensitivity of content is a rather coarse grained task. It will often result in the creation of multiple versions of a document, differing with respect to the levels of detail of the same information in order to accommodate the various levels of access rights. This approach is fraught with many problems:

- It leads to the duplication of information. The information that is shared across the entire range of access rights needs to be duplicated in every single version of the document.
- It can lead to multiple inconsistent documents. As the information covered in the document changes, depending on the level of update, all or some copies of the document will have to be updated. Any mistake or omission in this process will render the documents inconsistent with one another.
- The cost of maintaining such documents in terms of time and money is very high.

It is possible to avoid the above problems by tagging the content in a single document with multiple levels of access. Such tagged content can then be compared against the security policy of the organization in terms of the access privileges of the individuals and the rating of the access medium such as wireless LAN.

Emerging research in markup languages have made it possible for organizations to design and create their own customized markup applications using tagged content. Specifically, the Extensible Markup Language (XML) has made it easier to tag document content, store and transmit text and structural data both on and off the web, and pass information between systems that otherwise would lack interoperability. Unlike its predecessors, it does not

specify a tag set or semantics and allows the flexible development of user-defined document types. Since XML is so flexible and it makes it very easy to work with data, the potential for its application in document management and information security is promising.

B. THESIS SCOPE

The primary goal of this thesis is to provide a proof of concept of an access control technique that uses XML in a three tier client/server system architecture. The objectives of this research are to:

- Study and explore available XML technologies
- Study existing XML-based access control policy languages and evaluate integration options
- Study mechanisms and current network technology that can be used to determine whether the client is connecting from a wired or wireless network
- Design and develop a prototype incorporating the use of XML and the above mentioned mechanism that may be used to enhance the security to information being accessed across the network, particularly the wireless network

C. EXPECTED BENEFITS OF THE RESEARCH

Despite the advances in the 802.11 technology, improper security is still a shortcoming. Currently, wireless LAN equipment uses Media Access Control (MAC) address-based access control lists and an authenticator to control access. Given the ease with which MAC addresses can be spoofed from a wireless network and changed, alternative approaches are required to augment security. The features of XML can be harnessed to enhance regulating access to data and the medium over which that data is transmitted, improving overall security. The merits of this approach are clear in environments where varying levels of user privileges exists and there is a likelihood

sensitive information can be accessed over a wireless network.

D. THESIS ORGANIZATION

The thesis is organized as follows:

- Chapter II – XML
- Chapter III – Design Considerations for XML Solution
- Chapter IV – Development and Implementation of the Prototype
- Chapter V – Conclusion and Future Work

THIS PAGE INTENTIONALLY LEFT BLANK

II. XML

A. INTRODUCTION

1. What is XML

XML stands for Extensible Markup Language. Although referred to as a markup language, it is actually a meta language that describes other languages. It is extensible in the sense that developers can create their own tags in order to define and share information between applications and otherwise interoperable computing systems.

2. The Benefits of XML

XML was developed to address the problems and limitations of its predecessors, the Standard Generalized Markup Language (SGML) and the Hypertext Markup Language (HTML). The driving force behind its development was the need for a language that was more flexible, less complex and would also improve the functionality of changing web technologies.

Applicable to this thesis, there are two overarching benefits of XML. First, it has no specific application. Since it is extensible, it provides an adaptable means to identify information and restructure data to work with different applications. Not only is data content easily separated from presentation, developers can just as well as easily design their own markup language based on the XML standard, by creating their own customized tags and defining the structural relationship between them. There are many examples of these XML-based languages:

- MathML, a language for representing mathematical formulas and their presentation
- Scalable Vector Graphics (SVG), a language for describing two-dimensional graphics

- Extensible Business Reporting Language(XBRL), a language for describing financial reports
- Synchronized Multimedia Integration Language (SMIL), a language for creating multimedia presentations [Bradley 2003]

Second, when combined with its related technologies, XML is an even more powerful tool. Data from a single XML source file can be formatted for multiple distributions, via multiple distribution channels. The obvious benefit here is that content needs only to be written once and different versions can still be delivered with relative ease.

B. XML BASICS

There are several basic concepts to the XML specification. Some that are key to using and understanding XML are the following.

1. The XML Document

XML documents consist of three parts: the prolog, the document body and the epilog. The prolog and epilog are both optional. The prolog provides information about the document itself while the epilog contains any final comments or processing instructions. The document body contains the markup tags and the document's content in a hierarchical tree structure. The following are definitions of the basic building blocks of a document.

a. Attribute

A qualifier on an XML tag that provides additional information. For example, in the tag <TRACK length = "5:37">, length is the attribute, and "5:37" is its value.

b. Tag

A piece of text that describes a unit of data, or element in XML. It is surrounded by angle brackets. For this example, <TRACK>Confessions</TRACK>, <TRACK> is the start tag.

c. Element

This is a unit of XML data, delimited by tags. Besides data, an element can have child elements or they can be empty. For example, the title element below has two track elements. An element can occur more than once in a document. Modifying symbols are used to specify how many times an element can actually occur. Table 2 is a summary and description of these symbols.

```
<Title>
    <TRACK>Confessions</TRACK>
    <TRACK>Burn</TRACK>
</Title>
```

Modifying Symbols	**Description**
None	Must occur exactly 1 time
?	Allow zero or one occurrences
+	Allow one or more times
*	Allow zero or more times

Table 1. Modifying Symbols

d. CDATA

A predefined XML tag for "Character DATA" that says "don't interpret these characters", as opposed to

"Parsed Character Data" (PCDATA), in which the normal rules of XML syntax apply. CDATA sections are typically used to show examples of XML syntax.

e. Data

The contents of an element generally used when the element does not contain any subelements. In the example below “Confessions” is the data.

<TRACK>Confessions</TRACK>

Documents are categorized as either well-formed or valid. A well-formed document is syntactically correct and adheres to the XML specification. A valid document is a well formed document associated with what is called a Document Type Definition (DTD) or XML Schema. [Carey 2004]

2. Schema and DTD

Validation of an XML source document assures that the data values satisfy a set of rules. Documents can be validated with either a DTD or schema, and in some cases, both. These are typically separate files from the instance document and they define the structure, the allowable elements that a document may contain, and the order ikn which things have must occur.[1] Although similar in purpose, the DTD and schema differ, however, in their syntax, and the kind and power of constraints they impose on the document structure and content. Schemas provide more structure and data constraints than DTDs. They support many more data types with capabilities for detailed restrictions where users can even create customized data

[1] DTDs may be placed in the instance document as well in a separate file. External DTDs can be shared among different documents.

types. Additionally, with schemas, mixed content is handled fairly easily. DTDs, on the other hand, are more widely supported.

Document validation is important for several reasons. Of significant importance is that it reduces the likelihood of errors in a document, thus reducing the amount of programming necessary for error checking. Validating a document against a schema or a DTD requires the use of an XML parser.

3. Parsers

An XML parser is a processor capable of reading an XML document structure and providing access to its content to other applications. Parsers are classified as either validating or non-validating parsers. A non-validating parser checks a document for correct XML syntax and document structure while a validating parser enforces a certain DTD or schema in addition to checking for XML conformance. Parsers also differ in the way they read XML documents. There are two major types of parsers, Document Object Model (DOM) and Simple API for XML (SAX). The DOM parser builds a hierarchical data structure from the content of the document. The SAX parser, on the other hand, reads the content sequentially. Whenever a start or end tag is encountered, instead of placing it into a hierarchy, an event is sent to the application.

A parser is the key building block for every XML application. It is essential for the automatic processing of XML documents. Currently, there are many parsers that support the XML standard. Examples of such parsers include, Microsoft XML Parser(MSXML), Apache Xerces parser and Sun’s Java API for XML Processing(JAXP).

C. SECURING XML DOCUMENT CONTENT

The primary goal of information security is to ensure integrity, confidentiality and availability of information. Availability, access control, and confidentiality are among the terms usually associated with protecting information. Availability ensures authorized users are not prohibited access and information is not disclosed to unauthorized users. Integrity ensures that the information is not altered and requires some technique that also controls access. When considering using XML to supplement security, these properties are very applicable.

Since XML documents have a tree-like structure, different parts of the document can be protected in different ways. To achieve this, a fine-grained access control mechanism needs to be employed. One key technique for achieving such granularity is to define permissions in terms of components in the XML document. The issue now becomes how to specify the permissions. A permission can be generalized to be a set of actions that can be performed. Due to the hierarchial nature of the document tree, permissions can be filtered down the tree.[BOTHA 2004]

D. XML AND ACCESS CONTROL

Consider the structure of financial reports. Typically a narrative of the company's performance in qualitative terms is given followed by a table that shows the exact numbers and a detailed discussion based on those numbers. Access to the table and subsequent discussion is usually disallowed for individuals who do not have full authorization either because they did not pay for the report or simply because they have a lower authorization

level. Additionally, if this material was considered highly sensitive in nature, people who may otherwise be given access if they were on a more secure medium could be denied access if using wireless media. In order to use XML to enforce such fine grain control, while at the same time preserving efficiency, an access control mechanism is required. A summary of access control administration, taken from [Luo] is provided:

> XML access control in general has two aspects: access control models and enforcement mechanisms.

1. Access Control Models

Several authorization-based access control models exist. Among them is the authorization model where a specific authorization sheet is associated with each XML document/DTD that expresses the authorizations on the document. There is also the provisional model where specific provisional actions are specified and associated with a primitive action.[2] When a user makes an access request of a system, the system tells the user to take certain actions in order to be authorized. Examples of provisional actions are auditing, signature verification and encryption.

2. Enforcement Mechanisms

There are two types of enforcement mechanisms. The first is the view-based enforcement where the idea is to create and maintain a view for each authorized user that contains exactly the portion of the XML document that is explicitly authorized. The limitations of this approach are high storage costs and reduced scalability. A sometimes more suitable option is to let the XML engine

[2] A primitive action can be either read, write, create or delete.

enforce access control at the node level. The idea is to associate an access control list with each node of the XML document. Although view independent, the complexity of managing and maintaining the access control lists may neither be practical nor cost effective.

III. DESIGN CONSIDERATIONS FOR XML SOLUTION

A. XML DESIGN CONSIDERATIONS

1. Related Work

In October 2000, the Tokyo Research Laboratory released their work on an XML-based language, XML Access Control Language (XACL), aimed at providing a sophisticated access control mechanism that would add various security features to XML documents. The language would specify the security policies to be enforced on specific accesses to XML documents. [Hada 2000]

XACL is based on the provisional authorization model. It uses a subject-object-action-condition four-tuple format for specifying policy that can be bounded to a document at either the DTD or document level. Conceptually, the XML document consists of both content and policy. Using XACL, developers can write policy rules that determine access privileges for a particular document. [Hada 2000]

a. Subjects

A subject signifies a user who has all the right credentials. It encompasses identity and role where identity includes information about group or organization membership. It is specified as a subject element with three child elements uid, role and group.

```
<!ELEMENT subject (uid?, role*, group*)>
<!ELEMENT uid     (#PCDATA)>
<!ELEMENT role    (#PCDATA)>
<!ELEMENT group   (#PCDATA)>
```

Figure 1. Subject Specification

b. Objects

An object can signify a single element as well as a set of elements in the target XML document. It is specified as an <object> element that uses a "href" attribute for identification.

```
<!ELEMENT object EMPTY>
<!ATTLIST object href CDATA #REQUIRED>
```

Figure 2. Object Specification

c. Actions

There are four primitive actions: read, write, create and delete that represent the set of possible user privileges. An action is specified as an <action> element with the associated attributes, permission and name. The permission attribute value is either grant or deny. Since XACL is based on the provisional authorization model, provisional actions can be specified for a primitive action. These actions include auditing, logging and digital signature verification. The <provisional_action> element has two attributes, name and timing. Name specifies the name of the provisional action and timing specifies whether the provisional action is performed before or after the requested action.

```
<!ELEMENT action (provisional_action*)>
<!ATTLIST action name (read|write|create|delete) #REQUIRED
                 permission (grant|deny) #REQUIRED>
<!ELEMENT provisional_action (parameter*)>
<!ATTLIST provisional_action name CDATA #REQUIRED
                              timing (before|after) "after">
```

Figure 3. Action Specification

Table 2 provides a summary of the primitive actions.

Type	Parameters to be specified in access requests	Semantics
read	No	To read values of all child nodes except for elements of the target XML element (See Policy Specification for the propagation policy).
write	A string to be written	To replace all child text nodes by a single text node whose value is equal to the specified string.
create	A string to be created	To append the specified element to the end of the list of children of the target element.
delete	No	To delete the target element.

Table 2. Summary of Actions [From Hada 2000]

d. Conditions

Conditions make the policy rules more flexible such that access is granted if and when the condition is satisfied. Figure 4 provides the syntax.

```
<!ELEMENT condition (predicate|condition)*)>
<!ATTLIST condition operation(and|or|not) #REQUIRED
<!ELEMENT predicate (parameter*)>
<!ATTLIST predicate name CDATA #REQUIRED>
<!ELEMENT parameter ANY>
<!ATTLIST parameter name CDATA #IMPLIED>
<!ELEMENT function (parameter*)>
<!ATTLIST function name CDATA #REQUIRED>
```

Figure 4. Condition Specification [From Hada 2000]

e. Policy Specification

When a policy is bound to the document at the level of that specific document, it is encoded in <policy> element contained in that document. A <policy> element essentially maps objects to rules, subjects, actions and conditions. Figure 5 shows the policy specification.

```
<!ELEMENT policy  (property?,xacl*)>
<!ELEMENT xacl    (object+,rule+)>
<!ELEMENT rule    (acl)+>
<!ELEMENT acl     (subject*, action+, condition?)>
<!ELEMENT property (propagation?,conflict_resolution?,
                    default?)>
<!ELEMENT propagation EMPTY>
<!ATTLIST propagation read   (no|up|down) "down"
                      write  (no|up|down) "down"
                      create (no|up|down) "no"
                      delete (no|up|down) "up">
<!ELEMENT conflict_resolution EMPTY>
<!ATTLIST conflict_resolution read   (dtp|gtp|ntp) "dtp"
                              write  (dtp|gtp|ntp) "dtp"
                              create (dtp|gtp|ntp) "dtp"
                              delete (dtp|gtp|ntp) "dtp">
<!ELEMENT default EMPTY>
<!ATTLIST default read   (grant|deny) "deny"
                  write  (grant|deny) "deny"
                  create (grant|deny) "deny"
                  delete (grant|deny) "deny">
```

Figure 5. Policy Specification [From Hada 2000]

f. Authorization Architecture

The XACL architecture consists of two main modules known as the access evaluation module and the request execution module. An overview of the architecture is as follows: an access request containing the target element, a subject and an action is initiated. Upon submission, the access evaluation module is given access to the target document and its policy. The request is then evaluated according to that policy and determines whether access is granted or denied. The decision is executed in the request execution module. Figure 6 shows the authorization architecture.

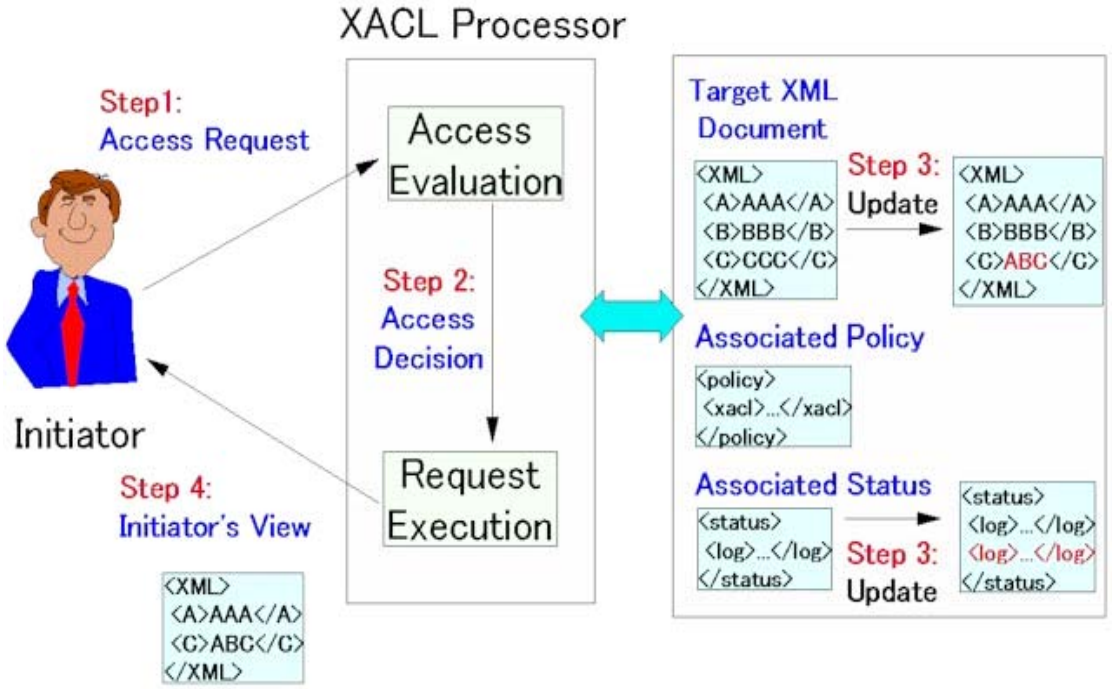

Figure 6. Authorization Architecture [From Hada 2000]

2. XML Policy File

Specifying access control data and a document's content in XML format can prove to be challenging. Currently, there are many access control policy languages to choose from, each with its own strengths and limitations. Since there is no one standard access control policy language, the choice of element tag names, their structure and sequence is left to preference and the specific application.

XACL's syntax and semantics were used as a reference for the format chosen as a solution for this thesis. Many choices had to be made, from deciding whether to bind the policy to each specific document to how each element is to

be represented. The syntax below is a sample of a policy that allows a user to read the content of an <stock> element if the <network> and <security> element values is equal to the user's access medium and rights, respectively. The next step is to transform the document for the user to view.

```
<portfolio>
  <stock>
    <network>wired</network>
    <security>high</security>
    <name symbol=></name>
    <earnings></earnings>
    <yield></yield>
</portfolio>
<policy>
  <xacl>
    <object href="/portfolio/stock"/>
      <rule>
        <acl>
          <action name="read" permission="grant"/>
          <condition operation="and">
            <predicate name="compareStr">
              <parameter>eq</parameter>
              <parameter>
                <function name="getValue"/>
                  <parameter>./network</parameter>
                </function>
              </parameter>
              <parameter>
                <function name="getNetwork"/>
              </parameter>
              <parameter>
                <function name="getValue"/>
                  <parameter>./security</parameter>
                </function>
```

```
                    </parameter>
                    <parameter>
                      <function name="getAccess"/>
                    </parameter>
                  </predicate>
                </condition>
              </acl>
            </rule>
          </xacl>
        </policy>
```

Figure 7. Sample Document

3. Transforming the Document

The document can be transformed into a variety of viewing formats. The Extensible Style Sheet Language (XSL) can transform the data into either a new XML document, HTML, portable document format(PDF) or rich text format(RTF). XSL has three parts:

- XSL-FO (Extensible Style sheet Language – Formatting Objects)
- XSLT (Extensible Style sheet Language Transformations)
- XPath

a. XSLT

XSLT is the part of XSL that actually translates the content into another presentation format or XML document. It uses an XSLT style sheet to convert the source document into the result document. A XSLT processor is required to complete the transformation of the source document. The processor reads both the source and the style sheet and applies the patterns described in the style sheet to the source document.

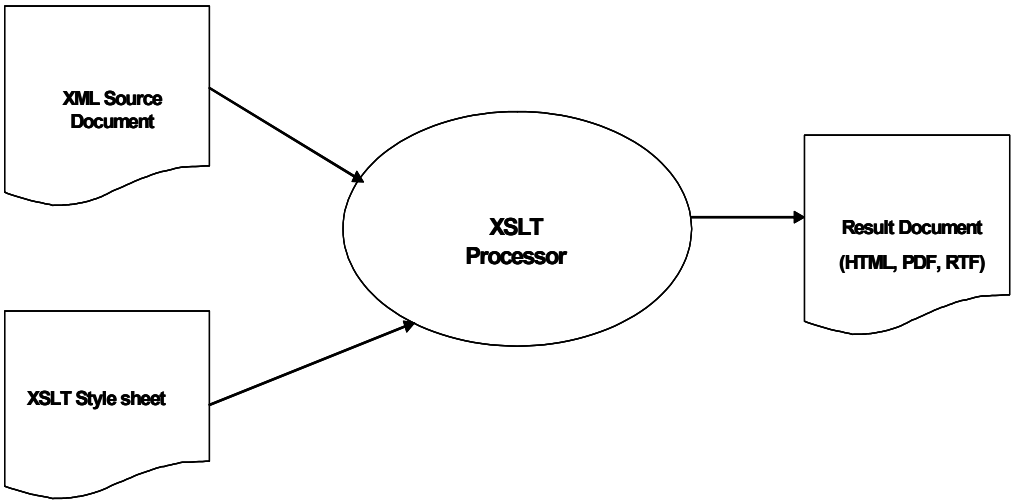

Figure 8. XSLT Transformation

b. Creating the XSLT Style Sheet

For this project, the XSLT style sheet transforms the source document into HTML. When developing a style sheet, the first logical step is to determine how the data is to appear on the web page. From there, the element and constructs needed to achieve the layout can be determined.

B. WIRELESS NETWORK DESIGN CONSIDERATIONS

Choosing an architecture for implementing networks is a significant one. The primary goal is to build an infrastructure that allows devices to communicate with each other. Most network designers are familiar with the key steps to ensure sufficient capacity and security for their wired networks. By their nature, wireless LANs are more difficult to design. The biggest challenge is closing the inherent security risks associated with wireless networking before exchanging data with the attached wired network. Most large infrastructure mode wireless networks combine the basic security available under IEEE 802.11 with 802.1X and Remote Authentication Dial-In User Service (RADIUS) support.

1. RADIUS Overview

RADIUS is a client/server protocol originally developed to enable centralized authentication, authorization and access control for dial-up remote access. Wireless access points, network access servers and other wireless networking devices now support the RADIUS protocol and use it to enforce authorization and authentication. With RADIUS, authentication and authorization occur simultaneously. Typically, the wireless client uses 802.11 to associate and send connection requests to the access point. The RADIUS capable access point then creates an Access-Request message with a username and encrypted password and forwards it to the RADIUS server. The RADIUS server verifies the user credentials stored in either a local users file or external database, and responds with Accept, Reject or Challenge. If accepted, the server sends an Access-Accept message that contains the parameters for the authorized connection to the access point. Parameters generally include the IP address to assign the user and the access list to be applied. Upon receipt of the Accept message, the access point completes the connection process with the wireless client in accordance with the services and parameters contained in the message. If rejected, the access points disassociates with the wireless client. The figure below is an illustration of the RADIUS architecture.

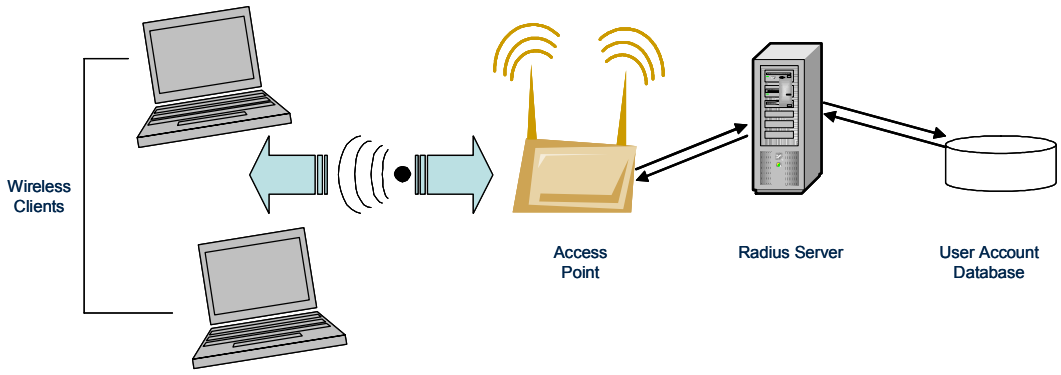

Figure 9. RADIUS Infrastructure

2. Practical Approaches for Wireless Security in Depth Up to the Application Level

Although extending applications to the wireless network reduces the level of security, practical approaches that address this issue can be applied. To achieve proper balance between secure communications and operational effectiveness in a wireless environment, certain requirements need to be met. Authenticating and encrypting the communications all the way to the specific wireless device is essential. Proof of identity should be obtained through two-way mutual authentication between the mobile user and the application/authentication server. Equally important and key, the solution of choice needs to be cost-effective, practical, appropriately simple for the mobile user, easily implemented and deployed and integrates well with the existing wired infrastructure. [V-ONE 2004]

A common approach when integrating a wireless network into an existing wired infrastructure is to connect the networks using either a gateway, a switch or a router. Using these devices, network managers can monitor and track the activities of wireless client devices and forward packets accordingly. In practice, most WLAN managers implement secure wireless gateways between their wireless

access points and the internal wired LAN. They add value by simplifying network management and are independent of access point brand or the wireless technology used. Often designed to support existing authentication methods such as a Windows Domain or RADIUS server, they enforce encryption, Quality of Service (QoS) and regulate access to network resources as defined by access policies.

WLAN gateways all work roughly in the same way. They function mostly at Layer 3, the networking layer. Layer 2 and below is processed at the access points and prior to reaching the gateway, all 802.11 specific properties are stripped and the 802.11 frames are converted to 802.3 Ethernet frames. Once converted to Ethernet, the fact that the frame originated from a wireless client becomes transparent and the only discerning information in the frame may be the source and destination addresses. The key then is to keep the wired and wireless traffic on separate subnets. Gateways, by their nature, are required to be on the same subnets at the access points they are managing. They can handle mobility between subnets as clients roam from one subnet to the next. Although using different subnets is a matter of choice, separating the subnets not only heightens security by making it easier to associate the access rights specific for each network, it also provides a simple means of identifying traffic destined for the “untrusted” wireless network.

With the separate subnet approach, individual access for each user to each application can be defined in terms of source address and made scalable down to the file level. An application can sit between the gateway and the file

that extracts the source address and determine if the file or any of its content is authorized to be sent.

C. OVERALL DESIGN AND ARCHITECTURE

For the overall architecture of the XML approach being researched, concepts from the Tokyo Research Project XACL were integrated into its design as well as the network design considerations addressed in the preceding section. For ease of management and security, the wired and wireless networks are maintained on different subnets. A client connects to the network either via the LAN or the wireless network. Upon verification of credentials, the client is given access to the network. When a client submits a request for a target XML document, the client's access privileges are determined from an integrated database. The client's source address and associated subnet mask is then evaluated to identify the medium being used to initiate the request. The access rights and access medium is passed to a module for access execution. This module processes the request in accordance with the target XML document's associated policy. A result document with the authorized content is created for the client. Figure 10 is a scenario for the overall architecture.

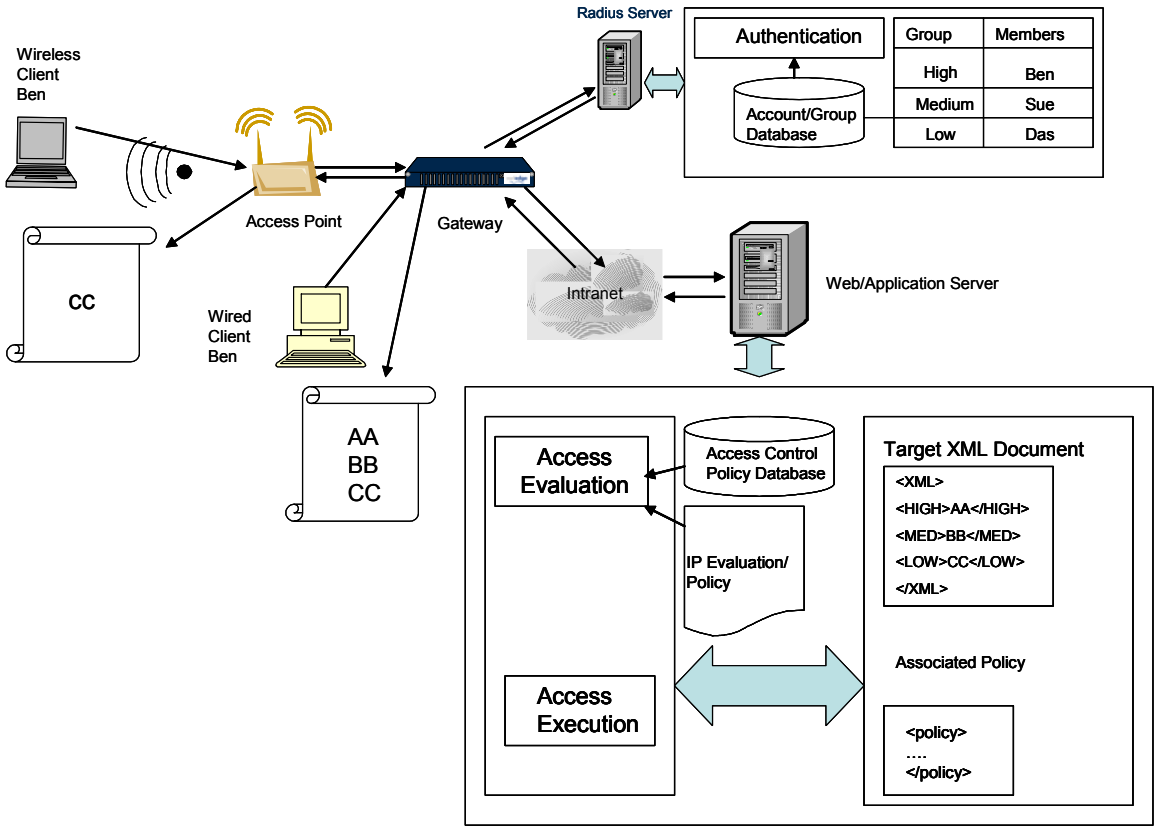

Figure 10. Overall Architecture Design [After Hada 2000]

THIS PAGE INTENTIONALLY LEFT BLANK

IV. DEVELOPMENT AND IMPLEMENTATION OF THE PROTOTYPE

A. DEVELOPMENT METHODOLOGY

Developing the prototype was very challenging. The following steps were taken in the prototyping process:

- Learn XML and assess design alternatives
- Study existing XML-based access control policy languages and evaluate integration options
- Design a sample XML document
- Design a Java based solution for user interface and document processing/parsing
- Integrate components to demonstrate concept

B. DEVELOPMENT TOOLS

1. XML SPY

Altova's XML Spy is a comprehensive tool that facilitates the generation of XML applications. XML Spy integrated-development environment (IDE) and the XML Spy Document Framework are bundled in XML Spy. The IDE is used to develop and manage XML documents, DTDs, schemas and XSLT style sheets. Developers can choose from one of four views while designing their documents. By default, XML Spy opens documents in the Enhanced Grid view. This view is suitable for structured editing. Figure 11 is a screen capture of the Enhanced Grid view.

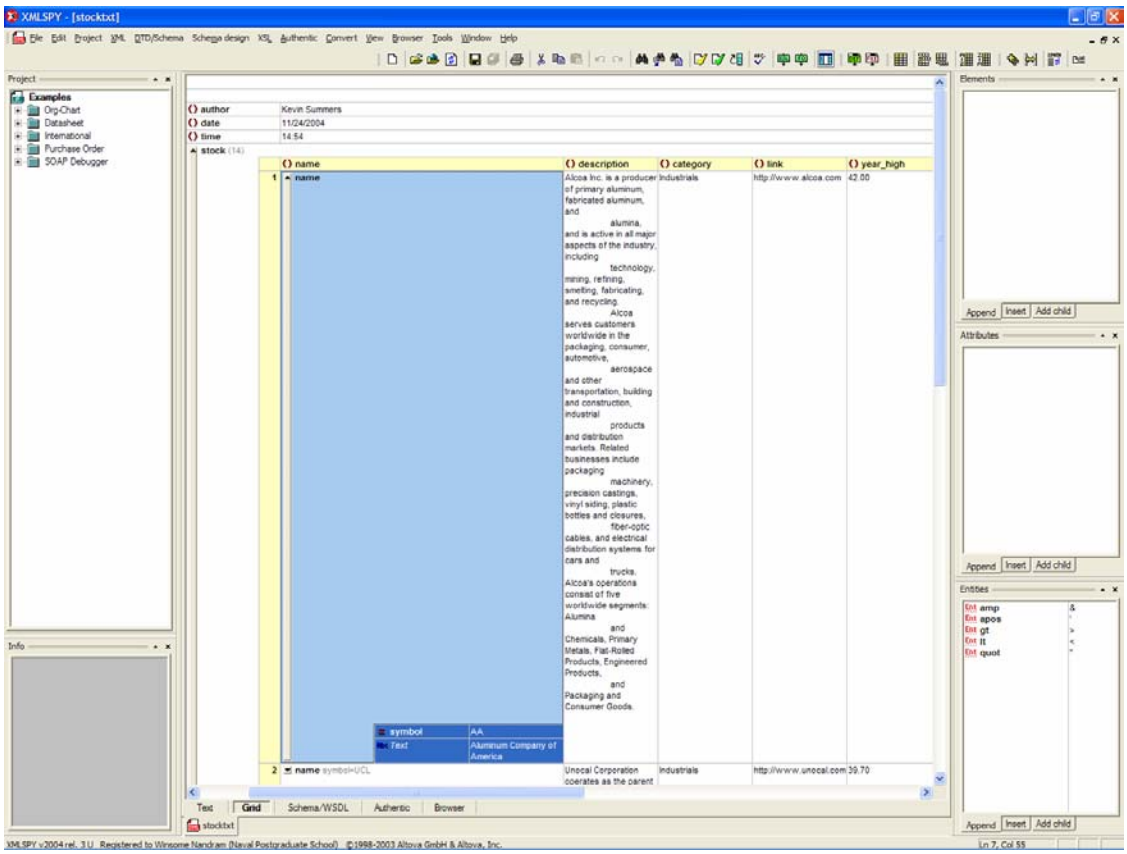

Figure 11. Screen Capture of Default Enhanced Grid View

For low-level work, there is a Text view with syntax-coloring (see Figure 12). The Database/Table view shows repeated elements in a tabular fashion and the integrated Browser View supports both CSS and XSL style-sheets.

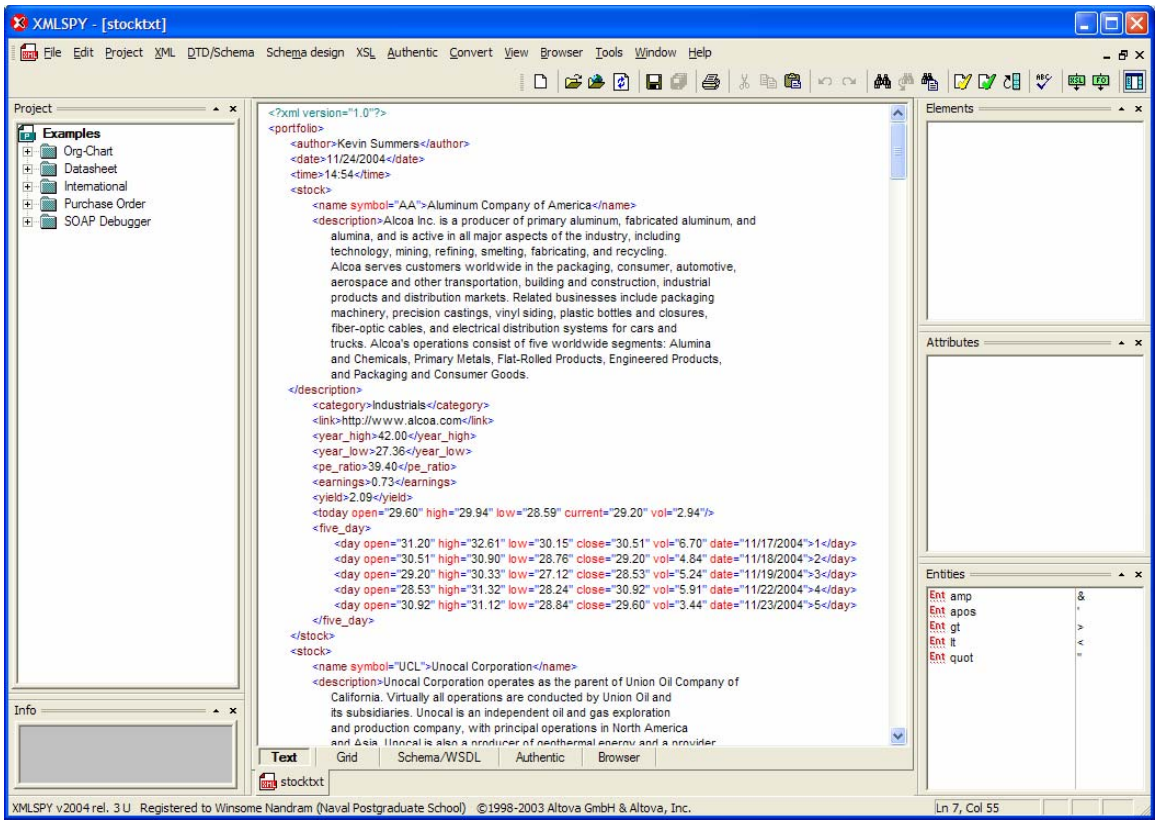

Figure 12. Screen Capture of Text View

The Document Framework contains the XSLT Designer and the XSLT Spy Form Editor. The XSLT Designer uses a drag and drop element interface to create a valid XSLT style sheet easily. The result can be displayed either in the XML Spy's embedded Internet Explorer view or sent out to the stand-alone browser view. XML Spy also features support for the MSXML XSLT parser.

2. MSXML

MSXML is one of the many XSLT processors used to load the contents of an XML file, apply the appropriate style sheet and display the results. It was first introduced in Internet Explorer 4.0 and has undergone several revisions since. MSXML versions 3.0 and 4.0 support the W3C DOM Level 2. Currently, only version 4.0 supports SAX and browser support for SAX is also limited.

W3C DOM was designed for both XML and HTML documents as an interface between a programming language and the document content. The MSXML DOM implementation actually has two interfaces which may be used. All the methods and properties of these interfaces are available when using MSXML from a scripting language such as JavaScript or VBScript. In this project, JavaScript is used to load the XML document and invoke the MSXML DOMDocument object.

3. Java API for XML Processing

There are many parsers available in today's marketplace. Sun, IBM and many other developers have implemented their own versions. They differ mainly by how well they support the XML standard. Java API for XML Processing (JAXP) is Sun's API for processing XML applications written in Java. JAXP supports both the DOM and SAX APIs and the XSLT standard. Designed for flexibility, it facilitates the use of any XML-compliant parser. Implementations offered by another vendor can be easily plugged in without altering source code.

JAXP uses the factory design pattern. The main JAXP APIs are defined in the javax.xml.parsers package that contains vendor-neutral factory classes--SAXParserFactory, DocumentBuilderFactory, and TransformerFactory. These classes are used to create their related objects, a SAXParser, a DocumentBuilder, and an XSLT transformer, respectively. DocumentBuilder, in turn, creates a DOM-compliant Document object.

JAXP offers a framework independent of the chosen parser. For this reason, JAXP's SAX parser was used for this thesis. The SAX API is ideal for reading and writing XML to a data repository or Web and for server-side and

high performance applications. In JAXP, the SAXParserFactory class is used to generate an instance of the SAX2 parser. When the parser's parse() method is invoked, the SAXReader object wrapped in the parser invokes one of several call back methods implemented in the application. The call back methods are defined by the SAX interfaces, ContentHandler, ErrorHandler, DTDHandler, and EntityResolver. Figure shows the basic outline of the JAXP SAX APIs. [JAXP 2004]

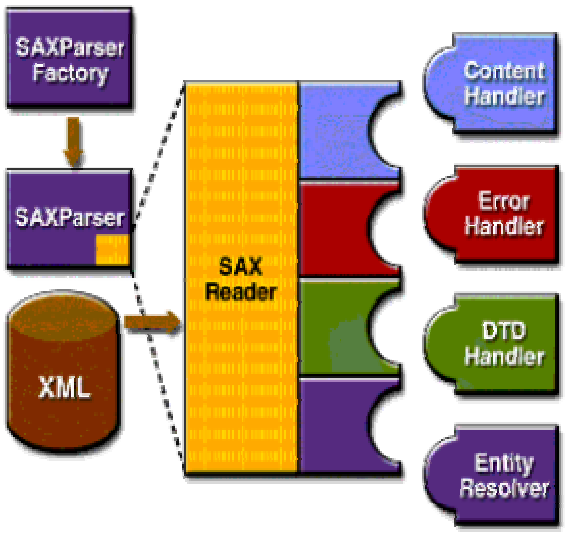

Figure 13. Outline JAXP SAX parsing APIs [From JAXP 2004]

To get started with JAXP, the Java Development Kit (JDK), the Xerces XML Parser and a Java Servlet Engine are needed.

4. Java Servlets

Servlets are Java programs that run in a server application to answer client requests. They are protocol and platform independent but are most commonly used with the HTTP protocol. Since Servlets are written in Java,

they have access to the entire family of Java APIs. They can do many tasks. Typical uses for Servlets are:

- Function as part of a middle tier in an enterprise network
- Accept form input and generate HTML pages dynamically
- Providing dynamic content, e.g. returning the results of a database query to the client.

Currenkly, servlets are a popular choice for building interactive Web applications. Third-party servlet containers are available for the Apache Tomcat Web Server, Microsoft IIS, and others. The following figure shows the basic processing model of a Java Servlet.

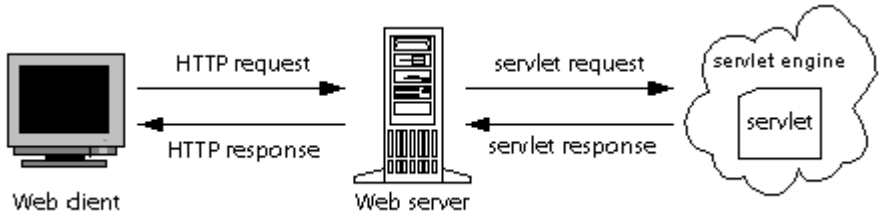

Figure 14. Processing Model of a Java Servlet [From Hall 2000]

C. SAMPLE XML FILES

1. Sample XML Document

The sample XML Document was inspired and derived from an example in [Carey 2004]. This example was useful since it explored the possibility of storing stock market information in XML format. More so, it facilitates the demonstration of the financial report scenario described in Chapter II, Section C.

During the course of the research, multiple formats for the document were designed. The formats differed primarily by the tag names and sequence chosen for the elements encoding the policy for the file. One format uses

an <access> element that tells the processor what should be done with the rest of the data contained in the document. The element is declared at the beginning of the document body and its value is one of three types assigned dynamically: high, medium, low. The other format encodes the elements in the document body with <high>, <medium> and <low> tags as appropriate for the classification of their data. The first made it easier for implementing an application that used XSL to transform the XML document to HTML. The latter is more effective for parsing, specifically with SAX. The sample documents are shown in Appendices A and B, respectively.

2. Sample XSLT Style Sheet

As is the case of the sample XML document, the sample XML style sheet, shown in Appendix C, was derived from an example in [Carey, 2004]. The style sheet is composed of several templates for each section of the document. The root template kicks off the transformation. Once the transformation begins, the value in the <access> element indicates to the XSLT processor which template to apply.

The XSLT processor of choice is MSXML. It is available from Microsoft and is contained in Internet Explorer 5.0 and above. Figure 15 is a screen shot of the result document for a user whose access privilege is “high” and is using the wired medium. Figure 16 represents the view for a user with “low” access privileges.

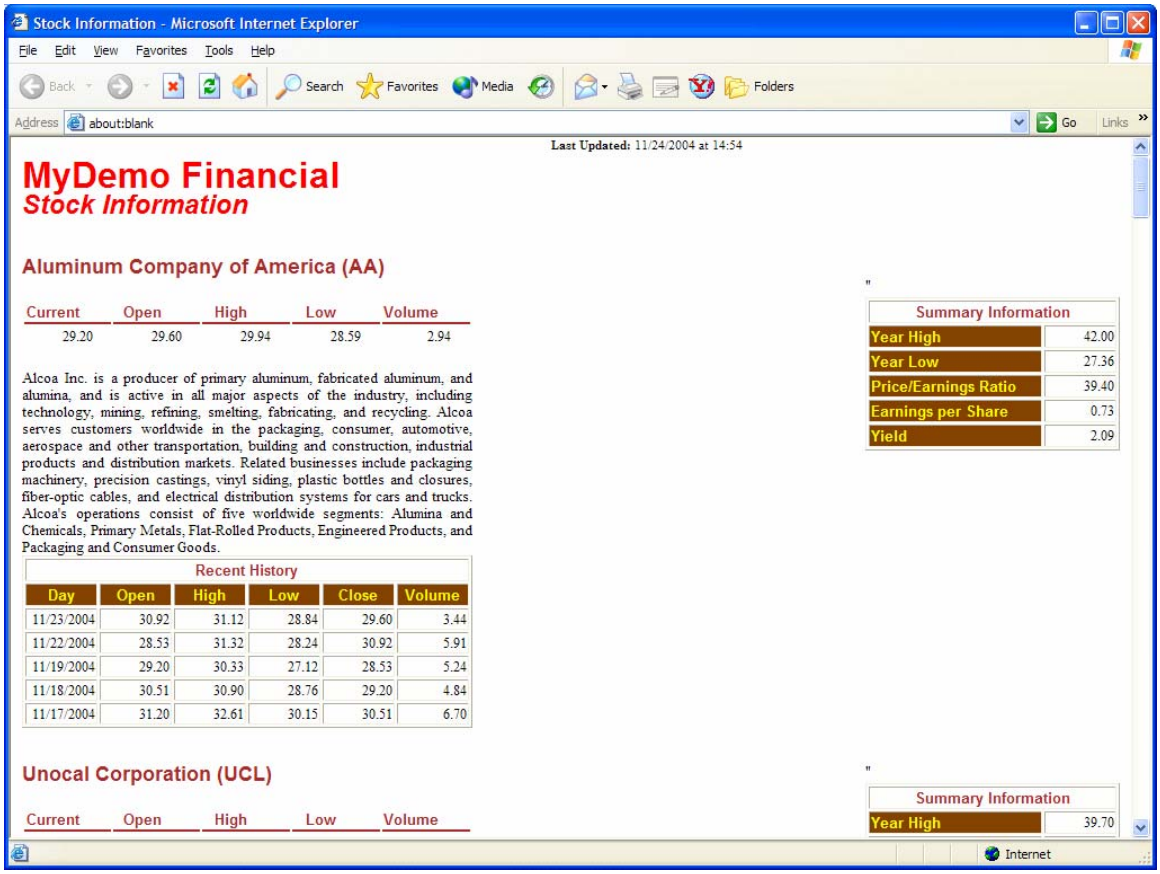

Figure 15. View for User with "High" Rating

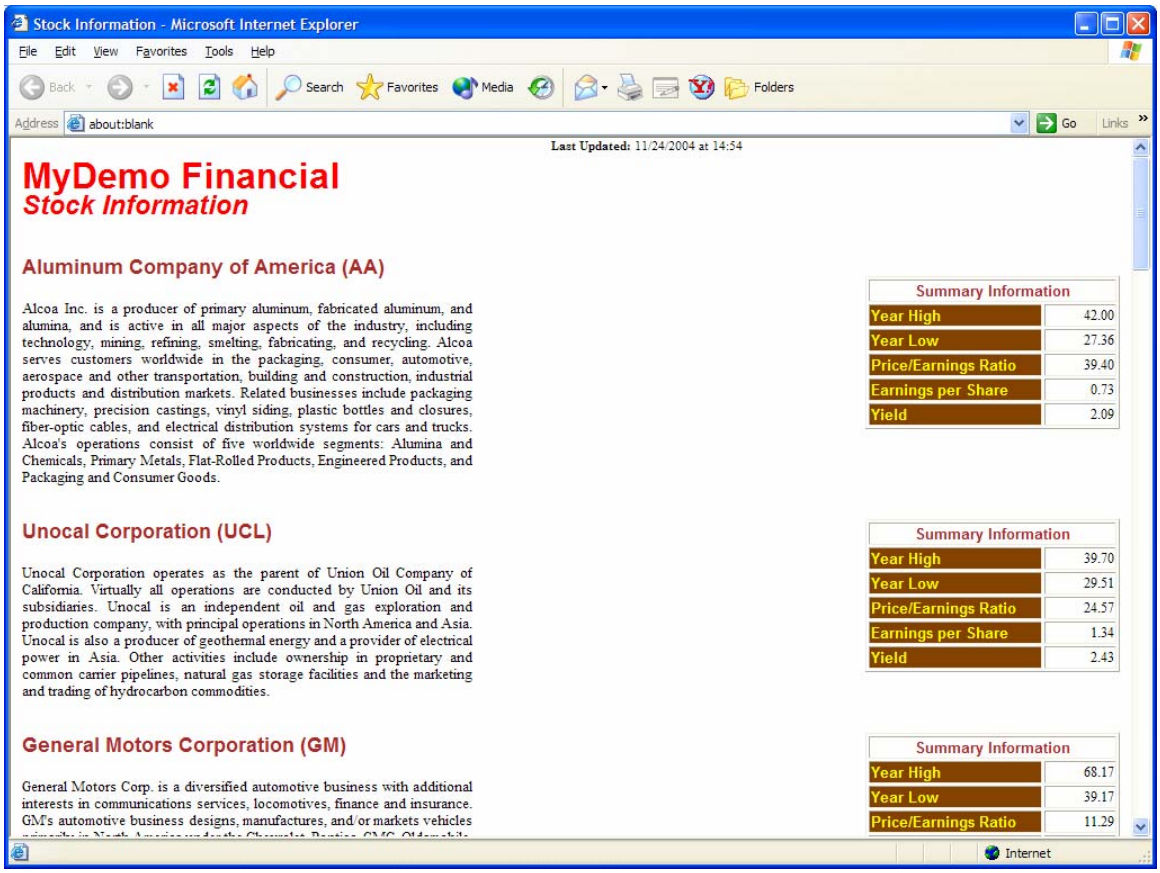

Figure 16. View for User with "low" Rating

D. DEMONSTRATION JAVA PROGRAM

Two application designs were implemented to demonstrate the concept of this thesis.

1. Option 1: Non Web-Based Application

This application consists of the second version of the stock XML Document and a Java program that sets up the user interface and as well as parse the document using SAX. SAX was chosen because it is an event-based parser that reacts to each element as they are encountered. In this case, when an element of interest is encountered, its content is processed appropriately for delivery to the client.

a. Design and Implementation

Five Java classes were developed for the Application; Login, Server, Handler, MyDemoParser and

XMLFileChooser. The Login, Server and Handler classes sets up and process client connections, authentication and authorization. When a client connects to the application server, the Login class displays a Login screen for the user. It then passes the user information to the Server and Handler classes for authentication and authorization. Figure 15 illustrates the Login GUI interface.

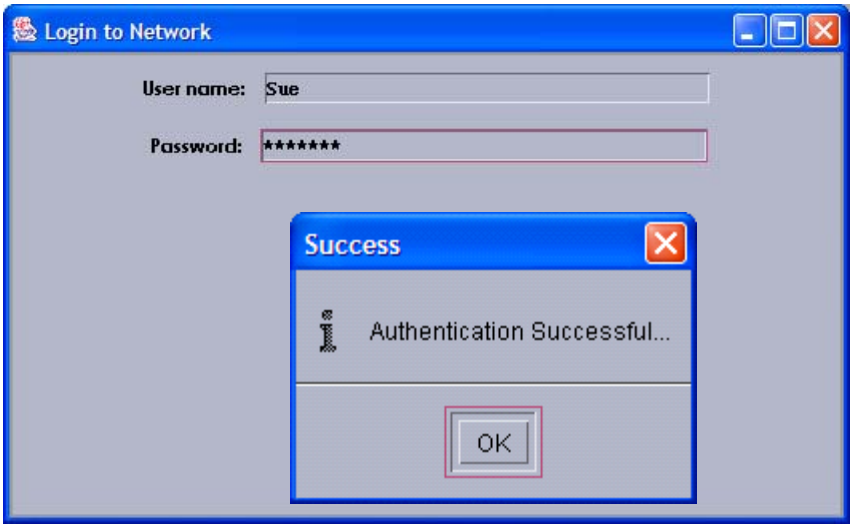

Figure 17. Login Screen

The ***public void checkAddress(byte [])*** method in the Handler class then checks the access media and combines it with the user's access privileges to determine the client's rating for that session. The user rating is then passed to the main class in the project, the MyDemoParser class. The purpose of this class is to parse the document and process its output. The classes in the org.xml.sax and javax.xml.parsers packages define all the interfaces used in this class. These interfaces are required for the SAX parser. The package, as well as the interfaces, are imported into the MyDemoParser class.

MyDemoParser obtains the name of the file to parse by displaying a fileChooser window from which the user may select the XML file. After the file is selected, the interface with the required SAX methods being used is implemented. The ContentHandler interface contains the methods that the SAX parser invokes in response to different parsing events. MyDemoParser extends the DefaultHandler class defined in the org.xml.sax. helpers package to implement the ContentHandler interface. DefaultHandler is simply an implementation of all the SAX handlers and provides empty methods for all the interface's events. The major methods are startDocument, endDocument, startElement, endElement and characters. MyDemoParser overrides the startElement, endElement and characters methods. When a start tag or end is encountered, the name of the tags is analyzed and the content processed accordingly. Figures 18 is an illustration of the fileChooser GUI.

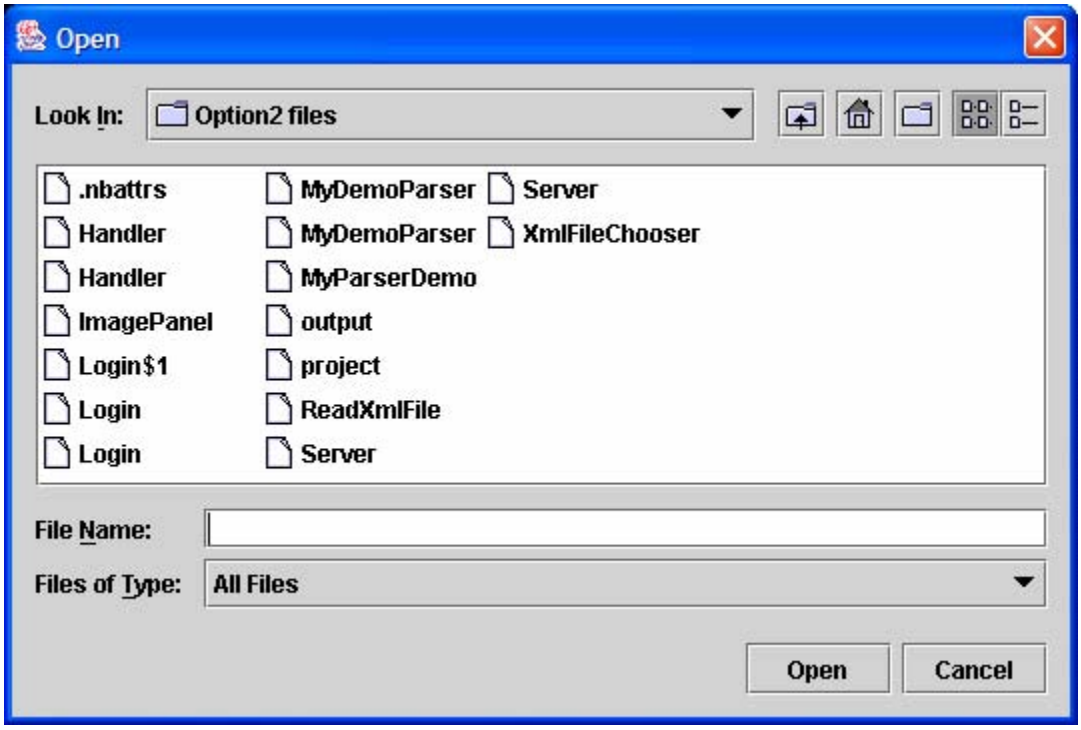

Figure 18. Demonstration FileChooser

2. Option 2: Web-Based Application

This alternate application was developed to exploit the benefits of web-based applications. Designs for web-based applications are usually classified as either client-side or server-side designs. Client side designs involve significant processing and presentation logic on the client side, while with server-side designs, the bulk of the processing is done on the server. In most modern corporate intranet infrastructure, the task is shared and the architecture reflects more of a medium weight client/server relationship.

For this application, the design is somewhere between the thin and medium weight design. HTML data is sent between the client and server. The client accesses data processed by the server through a servlet. The client uses the browser's built-in support to present the XML data to

the client. Internet Explorer version 5.0 is used to transform the XML data to HTML. The process begins with each user accessing the URL for the target XML document. The login web page is loaded for the user to input appropriate credentials. The information contained in the login form is sent to the web server using the HTTP post request. The web server, in turn, directs the request to the servlet engine for delivery to the servlet's doPost method. The servlet handles the HTTP post transaction, validates the user information, determines medium used for access and sends a response containing the client's rating to the web server. The web server delivers the response back to the client. In the process on the server side, the target XML document is loaded and referenced to a stylesheet that processes and transforms it to an HTML file. Figure 19 shows the components of the application.

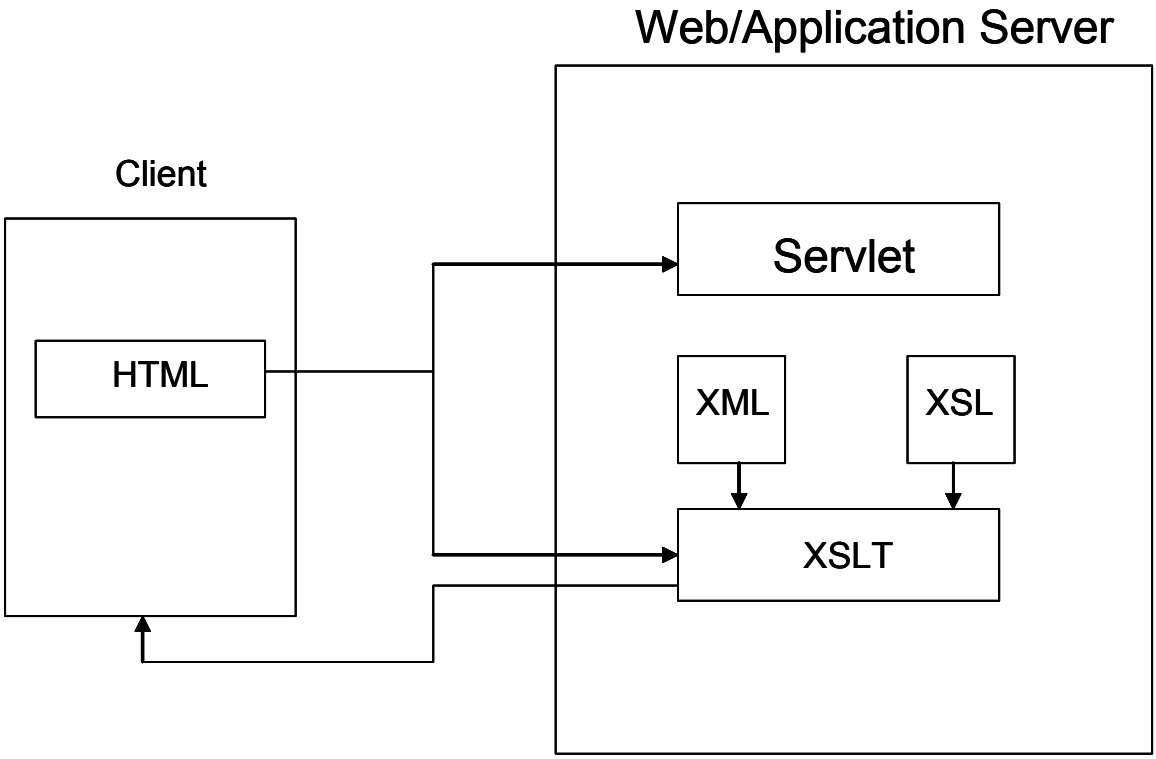

Figure 19. Application Architecture Components

This application consists of four files:

- Login.html

This is the user interface that sends the user's credentials to the web server for processing.

- NewServlet.java

A basic servlet that accepts POST submissions that contains user information. Within the servlet, the client is authenticated and given a rating.

- Finance.xml

Target file.

- Finance.xsl

XSLT style sheet.

E. SUMMARY

This chapter outlined the tools and architecture used to develop the prototype for this thesis. The XML document was successfully parsed and processed and the bare bones policy contained within the document was successfully integrated.

1. Advantages

The prototype produced the desired result, which was to restrict access to the document content based on a user's rating. Additional merits include:

- The prototype is a minimal design application that can be used as a foundation for a more dynamic solution.
- The application code is simple and makes good use of available technology including XML, Java and built-in browser facilities within the same system.
- During the developmental process, invaluable experience implementing XML processing, presentation and parsing was gained. Additionally, the exposure to 802.11 and wireless network design was extremely purposeful.

2. Limitations

- Although the prototype meets the current needs for proof of concept, it is not a viable solution for enterprise implementation.
- Data manipulation and output format is limited and is especially evident in the web-based solution.
- Access media determination depends largely on the assumption that the wired and wireless networks are on separate subnets and is limited to the examination of source IP addresses.

THIS PAGE INTENTIONALLY LEFT BLANK

V. CONCLUSION AND FUTURE WORK

A. CONCLUSION

Approaches for augmenting wireless security are not solely restricted to the typical solutions that make use of firewalls or traffic analyzers. The benefits of adopting and applying XML as a part of a security system can prove advantageous in many real life scenarios.

This thesis has researched and examined the major issues involved in pursing using XML for granular access control. Thorough research of the concepts and a wide variety of technology culminated with an overall proposed solution and a prototype that met the goals and scope of this thesis. Many lessons were learned during research and development.

XML has become the universal language for delivering platform-independent data and web content. DOM and SAX are the main APIs for XML. While DOM builds a data tree in memory for easier, non-sequential access to XML data, SAX provides a simpler way by reacting to data in the XML document at the moment that data is read. SAX clearly is more suited for fast, memory efficient parsing.

Als, Java and XML work well together to develop practical applications. The two technologies combined are well-suited for applications that provide different views of the same data to different users, which is the case for this prototype. XML provides the data for Java to process.

In summary, this thesis proves the concept that restricting access to content based on the medium of access heightens security. It also proves that such granular

access control can, in fact, be achieved using XML as an enabler. This novel approach not only exploits the best features of continuously developing technologies, it explores the issue of information security in a non-traditional way.

B. FUTURE WORK

Wireless security is a critical issue when managing network resources. On the positive side, there are some promising security products being developed. This thesis has given the basic framework for an approach using XML for access control. It presents opportunities for further research of the techniques and concepts used. The prototype can also be refined.

1. XML Access Control Language

Significant research is always being conducted on XML-based access control languages. The work of the IBM Tokyo Research Lab provides an excellent foundation for further work in this area.

2. Prototype

There are several shortcomings in the details of the prototype. The XML document does not truly reflect the framework presented. Additionally, it relies on a DTD for validation. Since schemas are more conducive to custom types, it should be considered for defining access permissions.

The development tools used for the prototype also require additional attention. This implementation was minimal and just enough to meet the requirements of this thesis. The framework is there for a comprehensive architecture that can support multiple formats for output.

C. RECOMMENDATIONS

XSL is one of many ways to apply a transformation on an XML document for web delivery. JavaServer Pages (JSP) technology can also be used to generate and transform XML documents. JSP technology is a Java technology with specifications for serving documents that combine dynamically generated and static markup language elements. It is ideally suited for working with XML as JSP pages may contain any type of text-based data and can leverage the Java programming language and APIs for processing XML data. A more abstract advantage of JSP is that it can be used in a more straightforward manner to use and generate documents that contain XML markup. A JSP page connects to a data source through a server-side object, transforms the information into data abstractions and then renders the data using JSP elements.

JSP, like XML can be processed and interpreted on any platform, to include handheld devices. It is the perfect companion for XML when used for web applications that require portability and share information. JSP and XML are two of the hottest buzzwords these days. An important future technology, JSP should definitely be considered when developing web applications.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A. SAMPLE XML DOCUMENT VERSION 1

The following is the sample XML document created for transformation to HTML.

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="finance.xsl" ?>
<portfolio>
    <access>high</access>
    <author>Kevin Summers</author>
    <date>11/24/2004</date>
    <time>14:54</time>
    <stock>
        <name symbol="AA">Aluminum Company of America</name>
        <description>Alcoa Inc. is a producer of primary aluminum, fabricated aluminum, and
            alumina, and is active in all major aspects of the industry, including
            technology, mining, refining, smelting, fabricating, and recycling.
            Alcoa serves customers worldwide in the packaging, consumer, automotive,
            aerospace and other transportation, building and construction, industrial
            products and distribution markets. Related businesses include packaging
            machinery, precision castings, vinyl siding, plastic bottles and closures,
            fiber-optic cables, and electrical distribution systems for cars and
            trucks. Alcoa's operations consist of five worldwide segments: Alumina
            and Chemicals, Primary Metals, Flat-Rolled Products, Engineered Products,
            and Packaging and Consumer Goods.
    </description>
        <category>Industrials</category>
        <link>http://www.alcoa.com</link>
        <year_high>42.00</year_high>
        <year_low>27.36</year_low>
        <pe_ratio>39.40</pe_ratio>
        <earnings>0.73</earnings>
        <yield>2.09</yield>
        <today open="29.60" high="29.94" low="28.59" current="29.20" vol="2.94"/>
        <five_day>
            <day open="31.20" high="32.61" low="30.15" close="30.51" vol="6.70" date="11/17/2004">1</day>
            <day open="30.51" high="30.90" low="28.76" close="29.20" vol="4.84" date="11/18/2004">2</day>
            <day open="29.20" high="30.33" low="27.12" close="28.53" vol="5.24" date="11/19/2004">3</day>
            <day open="28.53" high="31.32" low="28.24" close="30.92" vol="5.91" date="11/22/2004">4</day>
            <day open="30.92" high="31.12" low="28.84" close="29.60" vol="3.44" date="11/23/2004">5</day>
        </five_day>
    </stock>
    <stock>
        <name symbol="UCL">Unocal Corporation</name>
        <description>Unocal Corporation operates as the parent of Union Oil Company of
            California. Virtually all operations are conducted by Union Oil and
            its subsidiaries. Unocal is an independent oil and gas exploration
            and production company, with principal operations in North America
            and Asia. Unocal is also a producer of geothermal energy and a provider
            of electrical power in Asia. Other activities include ownership in
            proprietary and common carrier pipelines, natural gas storage facilities
            and the marketing and trading of hydrocarbon commodities.
    </description>
        <category>Industrials</category>
        <link>http://www.unocal.com</link>
        <year_high>39.70</year_high>
        <year_low>29.51</year_low>
        <pe_ratio>24.57</pe_ratio>
        <earnings>1.34</earnings>
        <yield>2.43</yield>
        <today open="33.35" high="34.00" low="32.66" current="33.80" vol="1.01"/>
        <five_day>
```

```
            <day open="33.91" high="34.26" low="31.80" close="32.12" vol="1.44" date="11/17/2004">1</day>
            <day open="32.12" high="33.94" low="31.55" close="33.41" vol="0.89" date="11/18/2004">2</day>
            <day open="33.41" high="33.82" low="32.45" close="32.94" vol="0.63" date="11/19/2004">3</day>
            <day open="32.94" high="33.31" low="30.40" close="31.24" vol="1.12" date="11/22/2004">4</day>
            <day open="31.24" high="33.64" low="31.00" close="33.35" vol="0.81" date="11/23/2004">5</day>
        </five_day>
    </stock>
    <stock>
        <name symbol="GM">General Motors Corporation</name>
        <description>General Motors Corp. is a diversified automotive business with
            additional interests in communications services, locomotives, finance
            and insurance. GM's automotive business designs, manufactures, and/or
            markets vehicles primarily in North America under the Chevrolet, Pontiac,
            GMC, Oldsmobile, Buick, Cadillac, Saturn and Hummer nameplates, and
            outside North America under the Opel, Vauxhall, Holden, Isuzu, Saab,
            Buick, Chevrolet, GMC and Cadillac nameplates. GM's communications
            services relate to its Hughes Electronics Corporation subsidiary,
            which includes digital entertainment, information and communications
            services, and satellite-based private business networks. GM also is
            engaged in the design, manufacturing and marketing of locomotives and
            heavy-duty transmissions. GM's financing and insurance operations are
            conducted primarily through General Motors Acceptance Corporation, which
            provides a broad range of financial services.
    </description>
        <category>Industrials</category>
        <link>http://www.gm.com</link>
        <year_high>68.17</year_high>
        <year_low>39.17</year_low>
        <pe_ratio>11.29</pe_ratio>
        <earnings>4.06</earnings>
        <yield>4.36</yield>
        <today open="46.67" high="47.00" low="44.53" current="46.67" vol="6.08"/>
        <five_day>
            <day open="49.24" high="49.50" low="47.28" close="48.81" vol="4.11" date="11/17/2004">1</day>
            <day open="48.81" high="49.20" low="46.50" close="47.10" vol="5.92" date="11/18/2004">2</day>
            <day open="47.10" high="48.44" low="46.25" close="48.11" vol="4.89" date="11/19/2004">3</day>
            <day open="48.11" high="48.25" low="46.81" close="47.24" vol="4.31" date="11/22/2004">4</day>
            <day open="47.24" high="47.94" low="46.20" close="46.67" vol="5.24" date="11/23/2004">5</day>
        </five_day>
    </stock>
    <stock>
        <name symbol="EK">Eastman Kodak Company</name>
        <description>Eastman Kodak Company (Kodak) is engaged primarily in developing,
            manufacturing and marketing traditional and digital imaging products,
            services and solutions for consumers, professionals, healthcare
            providers, the entertainment industry and other commercial customers.
            The Company is a major participant in the "infoimaging" industry,
            which is composed of devices (digital cameras and personal data
            assistants), infrastructure (online networks and delivery systems for
            images) and services and media (software, film and paper enabling
            people to access, analyze and print images). Kodak uses its technology,
            market reach and a host of industry partnerships to provide products
            and services for customers that need the information-rich content that
            images contain.
    </description>
        <category>Industrials</category>
        <link>http://www.kodak.com</link>
        <year_high>47.30</year_high>
        <year_low>24.40</year_low>
        <pe_ratio>na</pe_ratio>
        <earnings>-0.13</earnings>
        <yield>6.29</yield>
        <today open="29.40" high="29.84" low="27.83" current="29.84" vol="3.16"/>
        <five_day>
            <day open="31.05" high="31.15" low="28.76" close="29.24" vol="2.85" date="11/17/2004">1</day>
            <day open="29.24" high="29.94" low="27.24" close="28.51" vol="3.40" date="11/18/2004">2</day>
```

```
        <day open="28.51" high="30.44" low="28.06" close="29.20" vol="1.40" date="11/19/2004">3</day>
        <day open="29.20" high="30.55" low="28.21" close="29.40" vol="2.00" date="11/22/2004">4</day>
        <day open="29.40" high="30.94" low="29.10" close="29.40" vol="2.40" date="11/23/2004">5</day>
    </five_day>
</stock>
<stock>
    <name symbol="R">Ryder Systems Inc.</name>
    <description>Ryder System, Inc. is a provider of logistics, supply chain and
        transportation management solutions worldwide. The Company operates
        in three reportable business segments: Fleet Management Solutions,
        which provides full-service leasing, commercial rental and programmed
        maintenance of trucks, tractors and trailers to customers, principally
        in the United States, Canada and the United Kingdom; Supply Chain
        Solutions, which provides comprehensive supply chain consulting and
        lead logistics management solutions that support customers' entire
        supply chains, from inbound raw materials through distribution of
        finished goods throughout North America, in Latin America, Europe and
        Asia, and Dedicated Contract Carriage, which provides vehicles and
        drivers as part of a dedicated transportation solution, principally
        in North America.
</description>
    <category>Transportation</category>
    <link>http://www.ryder.com</link>
    <year_high>31.09</year_high>
    <year_low>17.02</year_low>
    <pe_ratio>13.79</pe_ratio>
    <earnings>1.85</earnings>
    <yield>2.35</yield>
    <today open="25.62" high="26.29" low="24.93" current="25.23" vol="0.26"/>
    <five_day>
        <day open="23.81" high="24.12" low="22.91" close="23.24" vol="0.48" date="11/17/2004">1</day>
        <day open="23.24" high="25.87" low="22.89" close="24.05" vol="0.18" date="11/18/2004">2</day>
        <day open="24.05" high="25.24" low="23.14" close="24.94" vol="0.31" date="11/19/2004">3</day>
        <day open="24.94" high="26.94" low="24.20" close="26.10" vol="0.61" date="11/22/2004">4</day>
        <day open="26.10" high="26.61" low="24.87" close="25.62" vol="0.35" date="11/23/2004">5</day>
    </five_day>
</stock>
<stock>
    <name symbol="ABF">Airborne Freight Corporation</name>
    <description>Airborne Freight Corporation is an air express company and air freight
        forwarder that expedites shipments of all sizes to destinations
        throughout the United States and most foreign countries. ABX Air, Inc.,
        the Company's principal wholly owned subsidiary, provides domestic
        express cargo service and cargo service to Canada. The Company is the
        sole customer of ABX for this service. ABX also offers limited charter
        service. Airborne Express provides door-to-door express delivery of
        small packages and documents throughout the United States and to and
        from most foreign countries. The Company also acts as an international
        and domestic freight forwarder for shipments of any size.
</description>
    <category>Transportation</category>
    <link>http://www.airborne.com</link>
    <year_high>23.34</year_high>
    <year_low>7.00</year_low>
    <pe_ratio>207.67</pe_ratio>
    <earnings>0.06</earnings>
    <yield>1.28</yield>
    <today open="12.80" high="13.18" low="11.90" current="13.00" vol="1.00"/>
    <five_day>
        <day open="10.61" high="11.67" low="10.20" close="11.05" vol="0.94" date="11/17/2004">1</day>
        <day open="11.05" high="12.15" low="10.84" close="11.94" vol="1.26" date="11/18/2004">2</day>
        <day open="11.94" high="12.03" low="10.91" close="11.20" vol="1.05" date="11/19/2004">3</day>
        <day open="11.20" high="11.82" low="10.94" close="11.41" vol="0.81" date="11/22/2004">4</day>
        <day open="11.41" high="12.91" low="11.06" close="12.80" vol="1.21" date="11/23/2004">5</day>
    </five_day>
</stock>
```

```
<stock>
    <name symbol="LUV">Southwest Airlines Co.</name>
    <description>Southwest Airlines Co. is a domestic airline that provides primarily
        short-haul, high-frequency, point-to-point, low-fare service. Southwest
        focuses principally on point-to-point, rather than hub-and-spoke service
        in markets with frequent, conveniently timed flights and low fares.
        The Company serves many conveniently located satellite or downtown
        airports such as Dallas Love Field, Houston Hobby, Chicago Midway,
        Baltimore-Washington International, Burbank, Manchester, Oakland,
        San Jose, Providence, Ft. Lauderdale/Hollywood and Long Island airports,
        which are typically less congested than other airlines' hub airports.
</description>
    <category>Transportation</category>
    <link>http://www.southwest.com</link>
    <year_high>22.00</year_high>
    <year_low>11.25</year_low>
    <pe_ratio>25.46</pe_ratio>
    <earnings>0.54</earnings>
    <yield>0.07</yield>
    <today open="14.00" high="14.59" low="13.53" current="14.00" vol="4.03"/>
    <five_day>
        <day open="13.06" high="14.21" low="12.87" close="13.81" vol="2.91" date="11/17/2004">1</day>
        <day open="13.81" high="14.76" low="13.75" close="14.14" vol="3.45" date="11/18/2004">2</day>
        <day open="14.14" high="16.05" low="13.91" close="15.27" vol="4.24" date="11/19/2004">3</day>
        <day open="15.27" high="15.65" low="14.00" close="14.91" vol="2.86" date="11/22/2004">4</day>
        <day open="14.91" high="15.20" low="13.87" close="14.00" vol="3.10" date="11/23/2004">5</day>
    </five_day>
</stock>
<stock>
    <name symbol="CNI">Canadian National Railway Co.</name>
    <description>Canadian National Railway is the only rail network on the continent
        to connect three coasts: the Pacific, the Atlantic and the Gulf of
        Mexico. The Company derives revenue from seven business units. The
        Petroleum and Chemicals unit transports a wide range of commodities,
        including chemicals, plastics, petroleum and gas products. The Metals
        and Minerals unit primarily transports nonferrous base metals, steel,
        equipment and parts. The Forest Products unit transports various types
        of lumber, panels, wood chips, wood pulp, pulpwood, printing paper,
        linerboard and newsprint. The Coal unit transports thermal and
        metallurgical grades of coal. The Grain and Fertilizer primarily
        transports commodities grown in western Canada and the United States
        Midwest. The Intermodal unit consists of a domestic segment and an
        international segment that transport consumer and manufactured goods.
        The Automotive unit is a carrier of automotive products originating
        in southwestern Ontario and Michigan.
</description>
    <category>Transportation</category>
    <link>http://www.cnrail.com</link>
    <year_high>53.75</year_high>
    <year_low>33.00</year_low>
    <pe_ratio>9.17</pe_ratio>
    <earnings>5.16</earnings>
    <yield>1.73</yield>
    <today open="47.70" high="48.14" low="46.63" current="47.19" vol="0.35"/>
    <five_day>
        <day open="51.75" high="52.47" low="49.76" close="51.24" vol="0.29" date="11/17/2004">1</day>
        <day open="51.24" high="51.94" low="49.24" close="50.35" vol="0.38" date="11/18/2004">2</day>
        <day open="50.35" high="50.76" low="48.10" close="49.24" vol="0.63" date="11/19/2004">3</day>
        <day open="49.24" high="50.24" low="47.75" close="48.50" vol="0.46" date="11/22/2004">4</day>
        <day open="48.50" high="48.79" low="47.10" close="47.70" vol="0.21" date="11/23/2004">5</day>
    </five_day>
</stock>
<stock>
    <name symbol="UNP">Union Pacific Co.</name>
    <description>Union Pacific Corporation operates primarily in the areas of rail
        transportation, through its indirect wholly owned subsidiary Union
```

```
            Pacific Railroad Company, and trucking, through its indirect wholly owned
            subsidiaries Overnite Transportation Company and Motor Cargo Industries,
            Inc. Union Pacific Railroad Company is a Class I railroad that operates
            in the United States, with over 33,000 route miles linking Pacific Coast
            and Gulf Coast ports to the Midwest and eastern United States gateways
            and several north/south corridors to key Mexican gateways. Overnite is
            a major interstate trucking company specializing in less-than-truckload
            (LTL) shipments. Motor Cargo is a western regional LTL carrier that
            provides service throughout 10 western states. The Company's other product
            lines are comprised of the corporate holding company, which largely
            supports the Railroad, Fenix LLC, affiliated technology companies and
            self-insurance activities.
</description>
    <category>Transportation</category>
    <link>http://www.up.com</link>
    <year_high>59.70</year_high>
    <year_low>58.15</year_low>
    <pe_ratio>14.32</pe_ratio>
    <earnings>4.06</earnings>
    <yield>1.38</yield>
    <today open="59.00" high="59.70" low="58.20" current="59.32" vol="1.81"/>
    <five_day>
        <day open="59.01" high="59.77" low="58.71" close="59.55" vol="1.07" date="11/17/2004">1</day>
        <day open="59.55" high="59.94" low="58.01" close="58.76" vol="1.55" date="11/18/2004">2</day>
        <day open="58.76" high="59.77" low="58.00" close="59.24" vol="1.76" date="11/19/2004">3</day>
        <day open="59.24" high="59.40" low="57.71" close="58.14" vol="2.24" date="11/22/2004">4</day>
        <day open="58.14" high="59.24" low="57.74" close="59.00" vol="1.20" date="11/23/2004">5</day>
    </five_day>
</stock>
<stock>
    <name symbol="ED">Consolidated Edison Co.</name>
    <description>Consolidated Edison, Inc. (Con Edison) is the holding company of
        Consolidated Edison Company of New York, Inc. (Con Edison of New York)
        and Orange and Rockland Utilities, Inc. (OR). Con Edison's principal
        business segments are the regulated electric, gas and steam businesses
        of its utility subsidiaries, and the unregulated businesses of its other
        subsidiaries. Con Edison of New York provides electric service in all
        of New York City (except part of Queens) and most of Westchester County,
        an approximately 660-square-mile service area with a population of more
        than eight million. It also provides gas service in Manhattan, The Bronx
        and parts of Queens and Westchester, and steam service in part of
        Manhattan, and its utility subsidiaries provide electric service in
        southeastern New York and in adjacent sections of New Jersey and
        northeastern Pennsylvania, an approximately 1,350 square mile service
        area.
</description>
    <category>Utilities</category>
    <link>http://www.conedison.com</link>
    <year_high>45.40</year_high>
    <year_low>35.50</year_low>
    <pe_ratio>11.76</pe_ratio>
    <earnings>3.16</earnings>
    <yield>5.98</yield>
    <today open="37.17" high="37.50" low="36.43" current="37.05" vol="1.22"/>
    <five_day>
        <day open="36.24" high="37.12" low="35.38" close="36.91" vol="1.46" date="11/17/2004">1</day>
        <day open="36.91" high="37.94" low="36.04" close="37.20" vol="1.14" date="11/18/2004">2</day>
        <day open="37.20" high="38.94" low="36.14" close="38.41" vol="1.21" date="11/19/2004">3</day>
        <day open="38.41" high="39.06" low="37.04" close="37.93" vol="1.31" date="11/22/2004">4</day>
        <day open="37.93" high="38.24" low="37.07" close="37.17" vol="1.06" date="11/23/2004">5</day>
    </five_day>
</stock>
<stock>
    <name symbol="AEP">American Electrical Power Company, Inc.</name>
    <description>American Electric Power Company, Inc. (AEP) is a public utility holding
        company that directly or indirectly owns domestic electric utility
```

```
            subsidiaries and varying percentages of other subsidiaries. The operating
            revenues of AEP and its subsidiaries are mostly derived from the marketing
            and trading of power and gas and the furnishing of electric service. The
            Company's operations are divided into three business segments: Wholesale,
            Energy Delivery and Other. The Wholesale Segment involves the generation
            of electricity for sale to retail and wholesale customers, the marketing
            and trading of electricity and gas worldwide, gas pipeline and storage
            services, and other energy supply related business. The Energy Delivery
            Segment is engaged in domestic electricity transmission and distribution.
            The Other Segment involves foreign electricity generation investments,
            foreign electricity distribution and supply investments, and
            telecommunication services.
    </description>
        <category>Utilities</category>
        <link>http://www.aep.com</link>
        <year_high>48.90</year_high>
        <year_low>33.02</year_low>
        <pe_ratio>12.98</pe_ratio>
        <earnings>2.74</earnings>
        <yield>6.75</yield>
        <today open="36.10" high="36.42" low="34.35" current="34.45" vol="2.67"/>
        <five_day>
            <day open="38.77" high="39.02" low="37.94" close="38.23" vol="2.41" date="11/17/2004">1</day>
            <day open="38.23" high="39.55" low="37.29" close="38.94" vol="2.91" date="11/18/2004">2</day>
            <day open="38.94" high="39.23" low="37.79" close="38.10" vol="3.89" date="11/19/2004">3</day>
            <day open="38.10" high="38.77" low="37.27" close="38.24" vol="2.34" date="11/22/2004">4</day>
            <day open="38.24" high="38.91" low="35.21" close="36.10" vol="2.12" date="11/23/2004">5</day>
        </five_day>
    </stock>
    <stock>
        <name symbol="PPL">PPL Corporation</name>
        <description>PPL Corporation is an energy and utility holding company. Through its
            subsidiaries, PPL generates electricity in power plants in the
            northeastern and western United States, markets wholesale or retail
            energy primarily in the northeastern and western portions of the
            United States and in Canada; delivers electricity to nearly six million
            customers in the United States, United Kingdom and Latin America, and
            provides energy services for businesses in the mid-Atlantic and
            northeastern United States. PPL is organized in segments consisting of
            Supply, Delivery and International. In addition, certain corporate
            service functionsare provided by PPL Services, an unregulated subsidiary
            of PPL.
    </description>
        <category>Utilities</category>
        <link>http://www.pplresources.com</link>
        <year_high>53.45</year_high>
        <year_low>28.54</year_low>
        <pe_ratio>11.10</pe_ratio>
        <earnings>2.69</earnings>
        <yield>4.82</yield>
        <today open="30.05" high="30.62" low="29.40" current="30.50" vol="0.84"/>
        <five_day>
            <day open="34.77" high="35.26" low="34.06" close="35.20" vol="1.38" date="11/17/2004">1</day>
            <day open="35.20" high="35.71" low="34.16" close="34.96" vol="0.94" date="11/18/2004">2</day>
            <day open="34.96" high="35.29" low="33.27" close="34.28" vol="1.41" date="11/19/2004">3</day>
            <day open="34.28" high="35.17" low="29.78" close="31.94" vol="1.61" date="11/22/2004">4</day>
            <day open="31.94" high="32.27" low="29.21" close="30.05" vol="1.24" date="11/23/2004">5</day>
        </five_day>
    </stock>
    <stock>
        <name symbol="MRO">Marathon Oil Co.</name>
        <description>Marathon Oil Corporation is engaged in the worldwide exploration and
            production of crude oil and natural gas; domestic refining, marketing
            and transportation of crude oil and petroleum products, primarily
            through its 62%-owned subsidiary, Marathon Ashland Petroleum LLC (MAP),
            and other energy-related businesses. Marathon is conducting exploration
```

```
            and development activities in 11 countries. Refining, marketing and
            transportation operations are primarily conducted by MAP and its
            subsidiaries, including its wholly owned subsidiaries, Speedway
            SuperAmerica LLC and Marathon Ashland Pipe Line LLC. Marathon also owns
            interest in various pipeline systems that were not contributed to MAP.
            Marathon, through its wholly owned subsidiary, Marathon Power Company,
            Ltd., pursues development, construction, ownership and operation of
            integrated gas and electric power projects in the global electrical
            power market.
        </description>
            <category>Industrials</category>
            <link>http://www.marathon.com</link>
            <year_high>32.75</year_high>
            <year_low>23.28</year_low>
            <pe_ratio>8.36</pe_ratio>
            <earnings>2.86</earnings>
            <yield>3.85</yield>
            <today open="24.15" high="24.63" low="23.70" current="24.08" vol="1.84"/>
            <five_day>
                <day open="24.95" high="25.38" low="23.65" close="24.32" vol="1.97" date="11/17/2004">1</day>
                <day open="24.32" high="24.95" low="23.71" close="24.16" vol="1.51" date="11/18/2004">2</day>
                <day open="24.16" high="26.25" low="23.87" close="25.87" vol="2.21" date="11/19/2004">3</day>
                <day open="25.87" high="26.13" low="24.02" close="24.78" vol="1.66" date="11/22/2004">4</day>
                <day open="24.78" high="25.02" low="23.95" close="24.15" vol="1.05" date="11/23/2004">5</day>
            </five_day>
        </stock>
        <stock>
            <name symbol="AZO">Autozone Co.</name>
            <description>AutoZone, Inc. is a specialty retailer of automotive parts and
                accessories, primarily focusing on do-it-yourself customers. Each
                auto parts store carries an extensive product line for cars, vans and
                light trucks, including new and re-manufactured automotive hard parts,
                maintenance items and accessories. The Company also has a commercial
                sales program in the United States that provides commercial credit and
                prompt delivery of parts and other products to local repair garages,
                dealers and service stations. AutoZone does not sell tires or perform
                automotive repair or installation. In addition, the Company sells
                automotive diagnostic and repair information software through its ALLDATA
                subsidiary, and diagnostic and repair information through
                alldatadiy.com.
        </description>
            <category>Industrials</category>
            <link>http://www.autozone.com</link>
            <year_high>84.50</year_high>
            <year_low>38.07</year_low>
            <pe_ratio>22.12</pe_ratio>
            <earnings>3.04</earnings>
            <yield>na</yield>
            <today open="68.00" high="69.30" low="65.80" current="67.19" vol="1.59"/>
            <five_day>
                <day open="66.87" high="67.20" low="65.89" close="66.55" vol="1.45" date="11/17/2004">1</day>
                <day open="66.55" high="68.16" low="66.00" close="67.20" vol="1.14" date="11/18/2004">2</day>
                <day open="67.20" high="69.10" low="67.06" close="68.16" vol="2.06" date="11/19/2004">3</day>
                <day open="68.16" high="70.24" low="67.77" close="69.72" vol="1.91" date="11/22/2004">4</day>
                <day open="69.72" high="69.91" low="67.12" close="68.00" vol="1.22" date="11/23/2004">5</day>
            </five_day>
        </stock>
    </portfolio>
```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B. SAMPLE XML DOCUMENT VERSION 2

The following is the sample XML document created for transformation by the SAX parser.

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="finance.xsl" ?>
<portfolio>
    <access>high</access>
    <author>Kevin Summers</author>
    <date>11/24/2004</date>
    <time>14:54</time>
    <stock>
        <low>
        <name symbol="AA">
            <low>Aluminum Company of America</low>
        </name>
        </low>
        <description>
            <low>Alcoa Inc. is a producer of primary aluminum, fabricated aluminum, and
            alumina, and is active in all major aspects of the industry, including
            technology, mining, refining, smelting, fabricating, and recycling.
            Alcoa serves customers worldwide in the packaging, consumer, automotive,
            aerospace and other transportation, building and construction, industrial
            products and distribution markets. Related businesses include packaging
            machinery, precision castings, vinyl siding, plastic bottles and closures,
            fiber-optic cables, and electrical distribution systems for cars and
            trucks. Alcoa's operations consist of five worldwide segments: Alumina
            and Chemicals, Primary Metals, Flat-Rolled Products, Engineered Products,
            and Packaging and Consumer Goods.</low>
        </description>
        <category>
            <low>Industrials</low>
        </category>
        <link>
        <low>http://www.alcoa.com</low>
        </link>
        <year_high>
        <low>42.00</low>
        </year_high>
        <year_low>
        <low>27.36</low>
        </year_low>
        <pe_ratio>
        <low>39.40</low>
        </pe_ratio>
        <earnings>
        <low>0.73</low>
        </earnings>
        <yield>
        <low>2.09</low>
        </yield>
        <high>
        <today open="29.60" high="29.94" low="28.59" current="29.20" vol="2.94"/>
        </high>
        <medium>
        <five_day>
            <day open="31.20" high="32.61" low="30.15" close="30.51" vol="6.70" date="11/17/2004">1</day>
            <day open="30.51" high="30.90" low="28.76" close="29.20" vol="4.84" date="11/18/2004">2</day>
            <day open="29.20" high="30.33" low="27.12" close="28.53" vol="5.24" date="11/19/2004">3</day>
            <day open="28.53" high="31.32" low="28.24" close="30.92" vol="5.91" date="11/22/2004">4</day>
            <day open="30.92" high="31.12" low="28.84" close="29.60" vol="3.44" date="11/23/2004">5</day>
```

```
        </five_day>
        </medium>
    </stock>
    <stock>
        <low>
        <name symbol="UCL">
            <low>Unocal Corporation</low>
        </name>
        </low>
        <description>
            <low>Unocal Corporation operates as the parent of Union Oil Company of
            California. Virtually all operations are conducted by Union Oil and
            its subsidiaries. Unocal is an independent oil and gas exploration
            and production company, with principal operations in North America
            and Asia. Unocal is also a producer of geothermal energy and a provider
            of electrical power in Asia. Other activities include ownership in
            proprietary and common carrier pipelines, natural gas storage facilities
            and the marketing and trading of hydrocarbon commodities.</low>
        </description>
        <category>
            <low>Industrials</low>
        </category>
        <link>
        <low>http://www.unocal.com</low>
        </link>
        <year_high>
        <low>39.70</low>
        </year_high>
        <year_low>
        <low>29.51</low>
        </year_low>
        <pe_ratio>
        <low>24.57</low>
        </pe_ratio>
        <earnings>
        <low>1.34</low>
        </earnings>
        <yield>
        <low>2.43</low>
        </yield>
        <high>
        <today open="33.35" high="34.00" low="32.66" current="33.80" vol="1.01"/>
        </high>
        <medium>
        <five_day>
            <day open="33.91" high="34.26" low="31.80" close="32.12" vol="1.44" date="11/17/2004">1</day>
            <day open="32.12" high="33.94" low="31.55" close="33.41" vol="0.89" date="11/18/2004">2</day>
            <day open="33.41" high="33.82" low="32.45" close="32.94" vol="0.63" date="11/19/2004">3</day>
            <day open="32.94" high="33.31" low="30.40" close="31.24" vol="1.12" date="11/22/2004">4</day>
            <day open="31.24" high="33.64" low="31.00" close="33.35" vol="0.81" date="11/23/2004">5</day>
        </five_day>
        </medium>
    </stock>
</portfolio>
```

APPENDIX C. XSL STYLE SHEET

Below is the XSL style sheet that transformed the XML document content for web viewing.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:fo="http://www.w3.org/1999/XSL/Format">
    <xsl:output method="html" version="4.0"/>
    <xsl:template match="/">
        <html>
            <head>
                <title>Stock Information</title>
                <link href="stock.css" rel="stylesheet" type="text/css"/>
            </head>
            <body>
                <div id="datetime">
                    <b>Last Updated: </b>
                    <xsl:value-of select="portfolio/date"/> at
                    <xsl:value-of select="portfolio/time"/>
                </div>
                <h1 class="title">MyDemo Financial</h1>
                <h2 class="title">Stock Information</h2>

                <xsl:apply-templates select="portfolio/stock"/>

            </body>
        </html>
    </xsl:template>

    <xsl:template match="stock">

            <table class="summary" border="1">
            <tbody>
                <tr>
                    <th colspan="2" class="summtitle">Summary Information</th>
                </tr>
                <tr>
                    <th class="summary">Year High</th>
                    <td class="number">
                        <xsl:value-of select="year_high"/>
                    </td>
                </tr>
                <tr>
                    <th class="summary">Year Low</th>
                    <td class="number">
                        <xsl:value-of select="year_low"/>
                    </td>
                </tr>
                <tr>
                    <th class="summary">Price/Earnings Ratio</th>
                    <td class="number">
                        <xsl:value-of select="pe_ratio"/>
                    </td>
                </tr>
                <tr>
                    <th class="summary">Earnings per Share</th>
                    <td class="number">
                        <xsl:value-of select="earnings"/>
```

```
                </td>
            </tr>
            <tr>
                <th class="summary">Yield</th>
                <td class="number">
                    <xsl:value-of select="yield"/>
                </td>
            </tr>
        </tbody>
        </table>

        <div class="stock_info">

        <xsl:choose>
            <xsl:when test="/portfolio/access= 'high' ">
            <xsl:apply-templates select="name"/>
            <xsl:apply-templates select="./today" />
            <p><xsl:value-of select="description"/></p>

            <xsl:apply-templates select="five_day" />
            </xsl:when>

            <xsl:when test="/portfolio/access= 'medium' ">
            <xsl:apply-templates select="./name"/>
            <p><xsl:value-of select="description"/></p>
            <xsl:apply-templates select="five_day" />
            </xsl:when>

            <xsl:when test="/portfolio/access= 'low' ">
            <xsl:apply-templates select="./name"/>
            <p><xsl:value-of select="description"/></p>
            </xsl:when>
        </xsl:choose>

        </div>


</xsl:template>

<xsl:template match="name">
     <h3 class="name">
        <xsl:value-of select="."/>
        (<xsl:value-of select="@symbol" />)
    </h3>
</xsl:template>

<xsl:template match="today">
<table class="today">
    <tbody>
        <tr>
            <th class="today">Current</th>
            <th class="today">Open</th>
            <th class="today">High</th>
            <th class="today">Low</th>
            <th class="today">Volume</th>

        </tr>
        <tr>
            <td class="number"><xsl:value-of select="@current" /></td>
            <td class="number"><xsl:value-of select="@open" /></td>
            <td class="number"><xsl:value-of select="@high" /></td>
            <td class="number"><xsl:value-of select="@low" /></td>
            <td class="number"><xsl:value-of select="@vol" /></td>
        </tr>
    </tbody>
```

```
    </table>
    </xsl:template>

    <xsl:template match="five_day">
    <table border="1" width="620" class="history">
        <tbody>
        <tr>
            <th class="histtitle" colspan="6">Recent History</th>
        </tr>
            <tr>
                <th class="history">Day</th>
                <th class="history">Open</th>
                <th class="history">High</th>
                <th class="history">Low</th>
                <th class="history">Close</th>
                <th class="history">Volume</th>
            </tr>
        </tbody>
        <xsl:apply-templates select="day">
            <xsl:sort data-type="number" order="descending" />
        </xsl:apply-templates>
    </table>
    </xsl:template>

    <xsl:template match="day">
    <tr>
        <td class="number"><xsl:value-of select="@date" /></td>
        <td class="number"><xsl:value-of select="@open" /></td>
        <td class="number"><xsl:value-of select="@high" /></td>
        <td class="number"><xsl:value-of select="@low" /></td>
        <td class="number"><xsl:value-of select="@close" /></td>
        <td class="number"><xsl:value-of select="@vol" /></td>
    </tr>
    </xsl:template>

</xsl:stylesheet>
```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX D. SOURCE CODE DEMONSTRATION SAX PARSER

```
/*
 * Program: MyDemoParser.java
 * Updated: August 25, 2004
 * Author:  Winsome Nandram
 * Purpose: This program parses an XML file and writes output to the screen.
 *          The XML file contains special tags that determines the output.
 *          Sample methods from the Sun website were used.
 *
 */

// created May 10, 2004

/**
 *  Class   :   MyDemoParser
 *  Purpose :   take an XML file and parse it using SAX
 *
 */

class MyDemoParser extends DefaultHandler implements Serializable
{
    StringBuffer textBuffer;
    String eName;
    String clearance, cl;
    String st;

        String err ="";
        FileReader input = null;
        BufferedReader r;

        FileWriter output = null;
        BufferedWriter w;
```

```
        //public constructor with no arguments

        public MyDemoParser() {

            clearance = "low";

        }

        /**
         *  Method  :   constructor MyDemoParser
         *  Purpose :   This method parses the file using registered SAX handlers
         *  Paramaters: String clear
         *
         */
        public MyDemoParser(String clear){

            setClearance(clear);
            cl = getClearance();

            // Display filechooser

            XmlFileChooser xmlFileChooser = new XmlFileChooser();

            try{
                input = new FileReader(xmlFileChooser.inFile);
                output   =   new   FileWriter("g:\\thesis   research\\program\\option2
files\\output.txt");

            } catch (IOException e) {
                err += e;
                JOptionPane.showMessageDialog(null, err);
                System.exit(0);
```

```
    }


    //Parse the input
    // Use an instance of ourselves as the SAX event handler
    DefaultHandler handler = new MyDemoParser();

    // Use the default (non-validating) parser
    SAXParserFactory factory = SAXParserFactory.newInstance();
    try {
        // Set up output stream
        out = new OutputStreamWriter(System.out, "UTF8");

        // Parse the input
        SAXParser saxParser = factory.newSAXParser();
        saxParser.parse( new File("finance2.xml"), handler);

    } catch (Throwable t) {
        t.printStackTrace();
    }
    System.exit(0);
}

private static Writer  out;
private String indentString = "   ";
private int indentLevel = 0;

//============================================================
// SAX DocumentHandler methods
//============================================================

/**
 *  Method  :   startDocument
```

```
 *  Purpose :   handles the events associated with the start of the document
 *  Parameters: none
 *
 **/

public void startDocument()
throws SAXException
{
    nl();
    nl();
    emit("START DOCUMENT");
    nl();
    emit("<?xml version='1.0' encoding='UTF-8'?>");
}

/**
 *  Method  :   endDocument
 *  Purpose :   indicated the end of the document
 *              handles the events associated with the end of the document
 *  Parameters: none
 *
 **/

public void endDocument()
throws SAXException
{
    nl();
    emit("END DOCUMENT");
    try {
        nl();
        out.flush();
    } catch (IOException e) {
        throw new SAXException("I/O error", e);
```

```
    }
}

/**
 *  Method  :   startElement
 *  Purpose :   handles and reports the occurence of an element
 *  Parameters: String namespaceURI, String sname, String qName
                Attributes attrs
 *  Throws  :   SAXException
 *
 **/

public void startElement(String namespaceURI,
                         String sName, // simple name
                         String qName, // qualified name
                         Attributes attrs)
throws SAXException
{
    indentLevel++;
    eName =sName;
    echoText();
    nl();
    if ("".equals(eName)) eName = qName; // not namespaceAware

}

/**
 *  Method  :   endElement
 *  Purpose :   handles events when the end of an element is reached
 *  Parameters: String namespaceURI, String sname, String qName
 *  Throws  :   SAXException
 *
 **/
```

```
            public void endElement(String namespaceURI,
                                   String sName, // simple name
                                   String qName  // qualified name
                                  )
            throws SAXException
            {
               if (sName.equals(clearance)){
                    echoText();
                }else {
                 textBuffer = null;
                 echoText();
                 indentLevel--;
               }
            }

            /**
             *  Method  :   characters
             *   Purpose :   handles characters between the elements and echos them to the
screen
             *  Parameters: char buf[], int offset, int len
             *  Throws  :   SAXException
             *
             **/

            public void characters(char buf[], int offset, int len)
            throws SAXException
            {
                if (textBuffer != null) {
                        nl();
                        echoText();

               }
```

```
    String s = new String(buf, offset, len);
    if (textBuffer == null) {
       textBuffer = new StringBuffer(s);
    } else {
       textBuffer.append(s);
    }
}

//============================================================
// Utility Methods ...
//============================================================

 /**
 *  Method  :   echoText
 *  Purpose :   display text accumulated in the character buffer
 *  Parameters: none
 *
 **/

private void echoText()
throws SAXException
{
    if (textBuffer == null) return;
    nl();
    String s = ""+textBuffer;
    if (eName.equals(clearance)){
    if (!s.trim().equals("")) emit(s);
   textBuffer = null;
    }
}

/**
 *  Method  :   emit
```

```
 *  Purpose :   outputs data
 *  Parameters: String s
 *  Throws  :   SAXException
 *              Wrap I/O exceptions in SAX exceptions, to
 *              suit handler signature requirements
 *
 */

private void emit(String s)
throws SAXException
{
    try {
        JOptionPane.showMessageDialog(null, s);
        out.write(s);
        out.flush();

    } catch (IOException e) {
        throw new SAXException("I/O error", e);
    }
}

/**
 *  Method  :   nl
 *  Purpose :   start a new line
 *  Parameters: none
 *  Throws  :   SAXException
 *
 **/
private void nl()
throws SAXException
{
    String lineEnd =  System.getProperty("line.separator");
    try {
```

```
            out.write(lineEnd);
            for(int i=0; i<indentLevel; i++) out.write(indentString);

        } catch (IOException e) {
            throw new SAXException("I/O error", e);
        }
    }

    /**
     *  Method  :   setClearance
     *  Purpose :   sets user security rating
     *  Parameters: String clearance
     *
     */

    public void setClearance(String clearance)
    {
        this.clearance = clearance;
        System.out.println("this is clearace value in setmethod:" + clearance);
    }

    /**
     *  Method  :   getClearance
     *  Purpose :   gets user security rating
     *  Parameters: none
     *
     */
        public String getClearance()
    {
        System.out.println("this is clearace value in get method:" + clearance);
        return clearance;
    }
}
```

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

[Apache 2004] The Apache XML Project. Xerces Java Parser Readme, http://xml.apache.org/xerces-j/index.html, date accessed August 31, 2004.

[Arbaugh 2003] Arbaugh, W. and Housley, R., "Security Problems in 802.11-Based Networks," Communications of the ACM, Volume 46, Number 5, p. 32, May 2003.

[Botha 2004] Botha, R. and Eloff, J., "An Access Control Architecture for XML Documents in Workflow Environments," http://www.ptech.ac.za/secwflow/images/papers/SAICSIT2001.pdf date accessed August 18, 2004.

[Bradley 2003] Bradley, G., "Introduction to Extensible Markup Language (XML) with Operations Research Examples," Newsletter of the INFORMS Computing Society, Volume 24, Number 1, p. 9, Spring 2003.

[Carey 2004] Carey, P., New Perspectives on XML-Comprehensive, 2004, Course Technology.

[Hada 2000] Hada, S. and Kudo, M., "XML Access Control Language: Provisional Authorization for XML Documents," http://www.tril.ibm.com/projects/xml/xacl/xacl-spec.html, date accessed August 18, 2004, Tokyo Research Laboratory, IBM Research, October 16, 2000.

[Hall 2000] Hall, Marty, Core Servlets and JavaServer Pages, 2000, Sun Microsystems Press, Upper Saddle River, New Jersey.

[JAXP 2004] http://java.sun.com/j2ee/1.4/docs/tutorial/doc/JAXPIntro.html date accessed August 18, 2004.

[V-ONE 2004] V-ONE CEO. "Open for Business – with Secure Mobile Communication," http://www.v-one.com/docs/ceo-agenda.pdf, date accessed August 31, 2004.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. Marine Corps Representative
   Naval Postgraduate School
   Monterey, California

4. Director, Training and Education, MCCDC Code C46
   Quantico, Virginia

5. Director, Marine Corps Research Center, MCCDC, Code C40RC
   Quantico, Virginia

6. Marine Corps Tactical Systems Support Activity (Attn: Operations Officer)
   Camp Pendleton, California

7. Professor Gurminder Singh
   Naval Postgraduate School
   Monterey, California

8. Academic Research Associate Arijit Das
   Naval Postgraduate School
   Monterey, California